US006099320A

United States Patent [19]

Papadopoulos

[11] **Patent Number: 6,099,320**

[45] **Date of Patent: Aug. 8, 2000**

[54] **AUTHORING SYSTEM AND METHOD FOR COMPUTER-BASED TRAINING**

[76] Inventor: **Anastasius Papadopoulos**, 721 S. Lieser Rd., Vancouver, Wash. 98664

[21] Appl. No.: **09/110,628**

[22] Filed: **Jul. 6, 1998**

Related U.S. Application Data

[60] Provisional application No. 60/052,792, Jul. 7, 1997.

[51] **Int. Cl.**[7] **G06T 7/20**

[52] **U.S. Cl.** **434/322**; 345/418; 345/302; 434/118; 434/323; 434/365

[58] **Field of Search** 434/118, 322, 434/323, 324, 327, 335, 365; 345/418, 473, 121, 425, 302; 703/13, 21, 22

[56] **References Cited**

U.S. PATENT DOCUMENTS

5,784,546 6/1998 Benman, Jr. 345/302

Primary Examiner—Jessica J. Harrison
Assistant Examiner—Chanda Harris
Attorney, Agent, or Firm—Fulbright & Jaworski, LLP

[57] **ABSTRACT**

The present invention is an integrated system for authoring, operating, maintaining, and administering a computer-based training program. The system includes integrated hardware and software that allows an author to create computer-based training modules that include audio, video, and textual content without requiring any programming. The invention uses a virtual training center metaphor that serves as the basis of the interface between the users and the system. The virtual training center provides a nonintimidating visual environment that is open and relaxing, thereby preparing the student to learn rapidly. To create a non-intimidating environment and to fix more of the student's attention on the subject matter to be learned, rather than on the learning tools themselves, the interface maintains as much as possible the illusion of being in a real training center. The system performs administrative functions, such as ensuring that students take courses in proper sequence, tracking training activity and course completion, and notifying the course administrator when a course is completed.

11 Claims, 15 Drawing Sheets

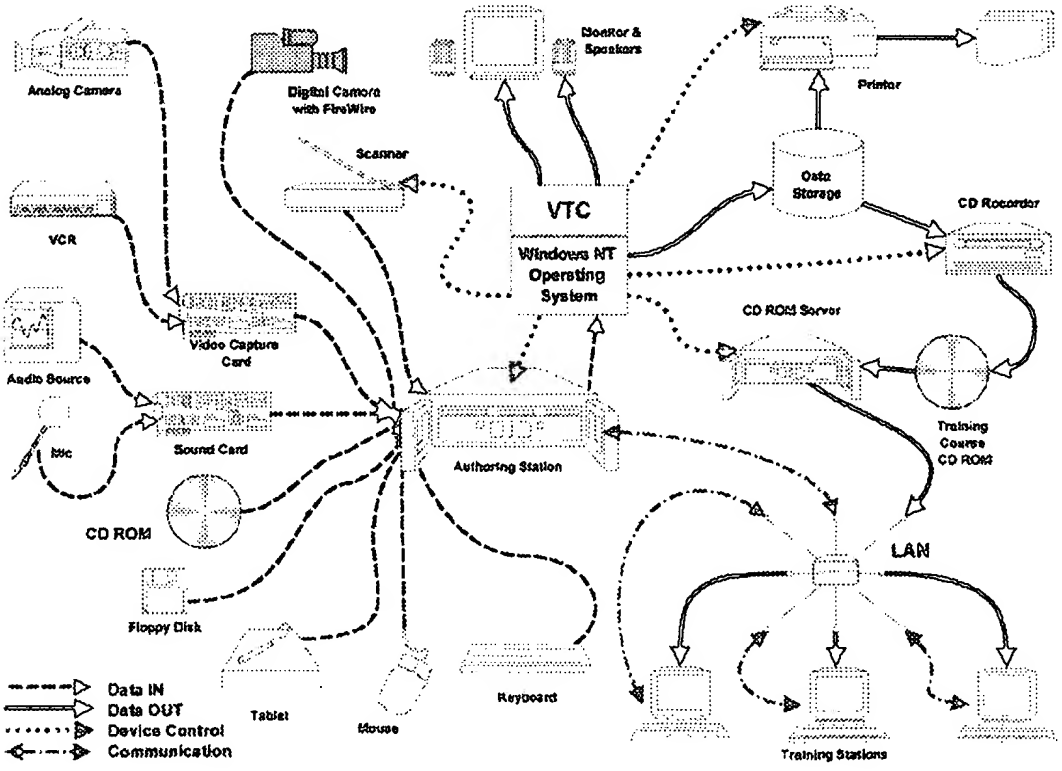


FIG. 1

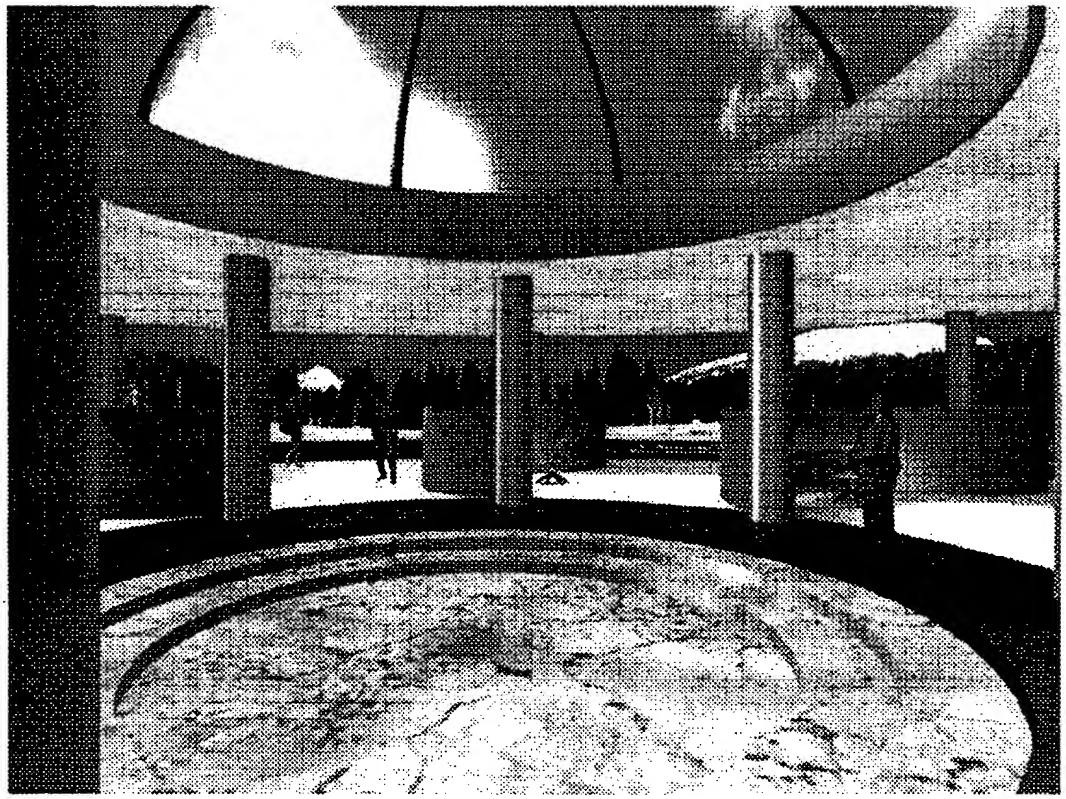


FIG. 2

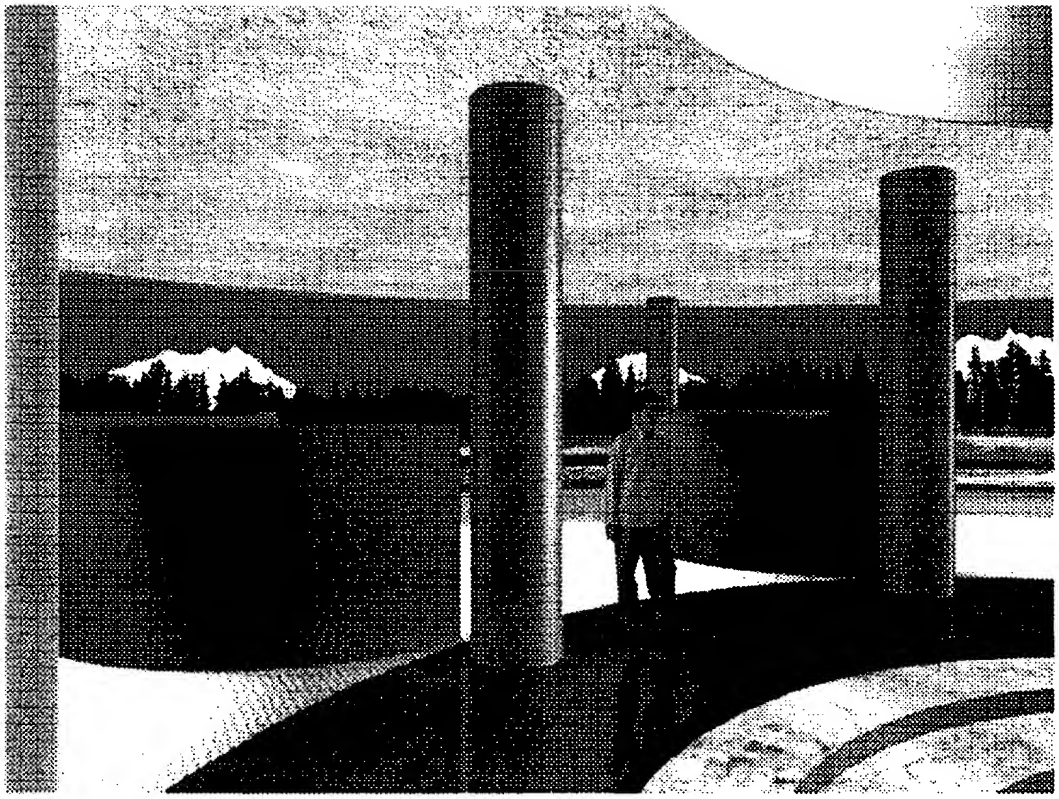


FIG. 3

FIG. 4

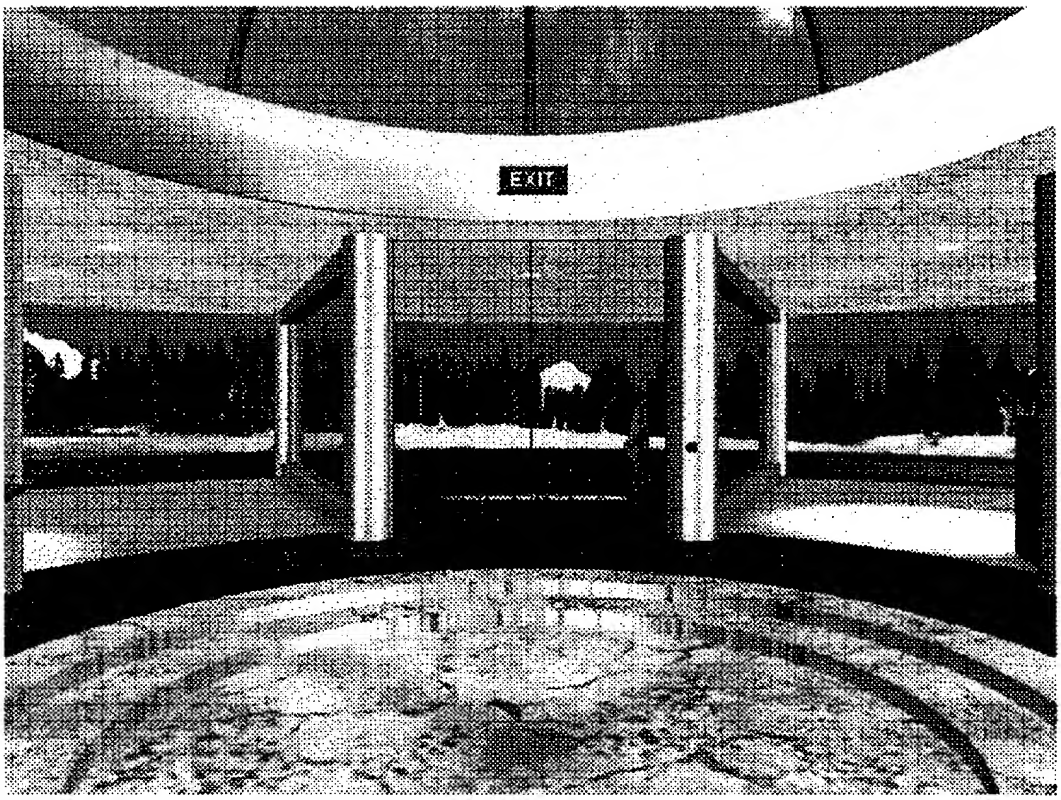


FIG. 5

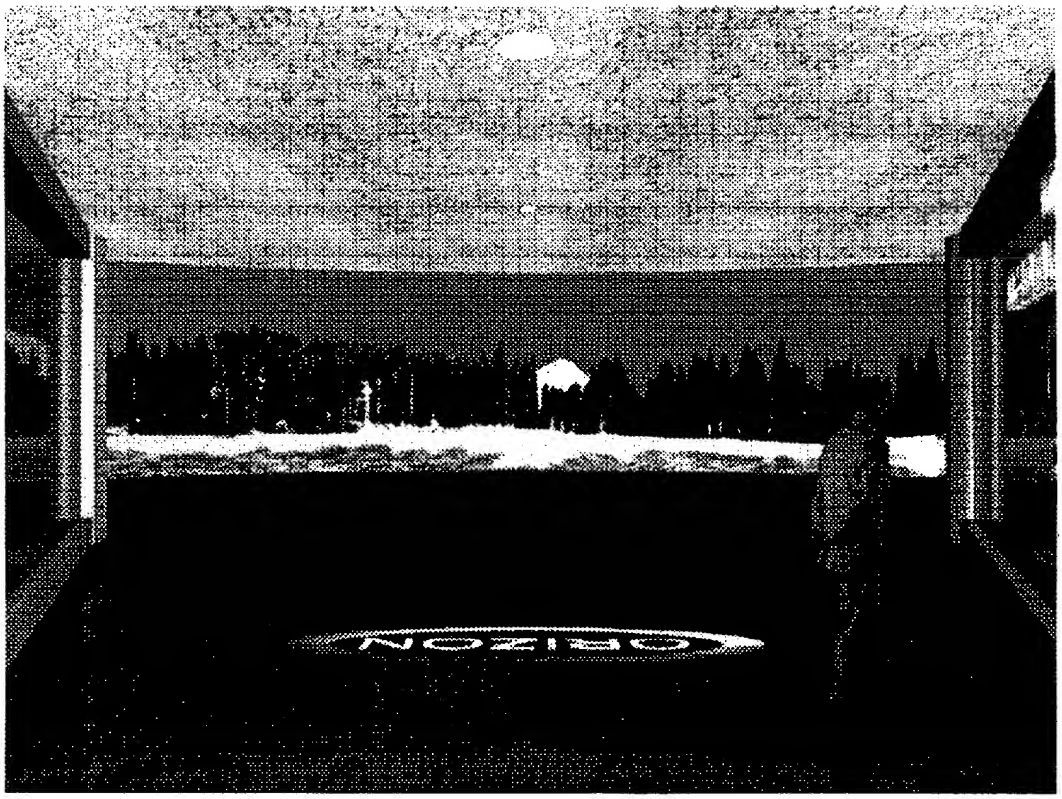


FIG. 6

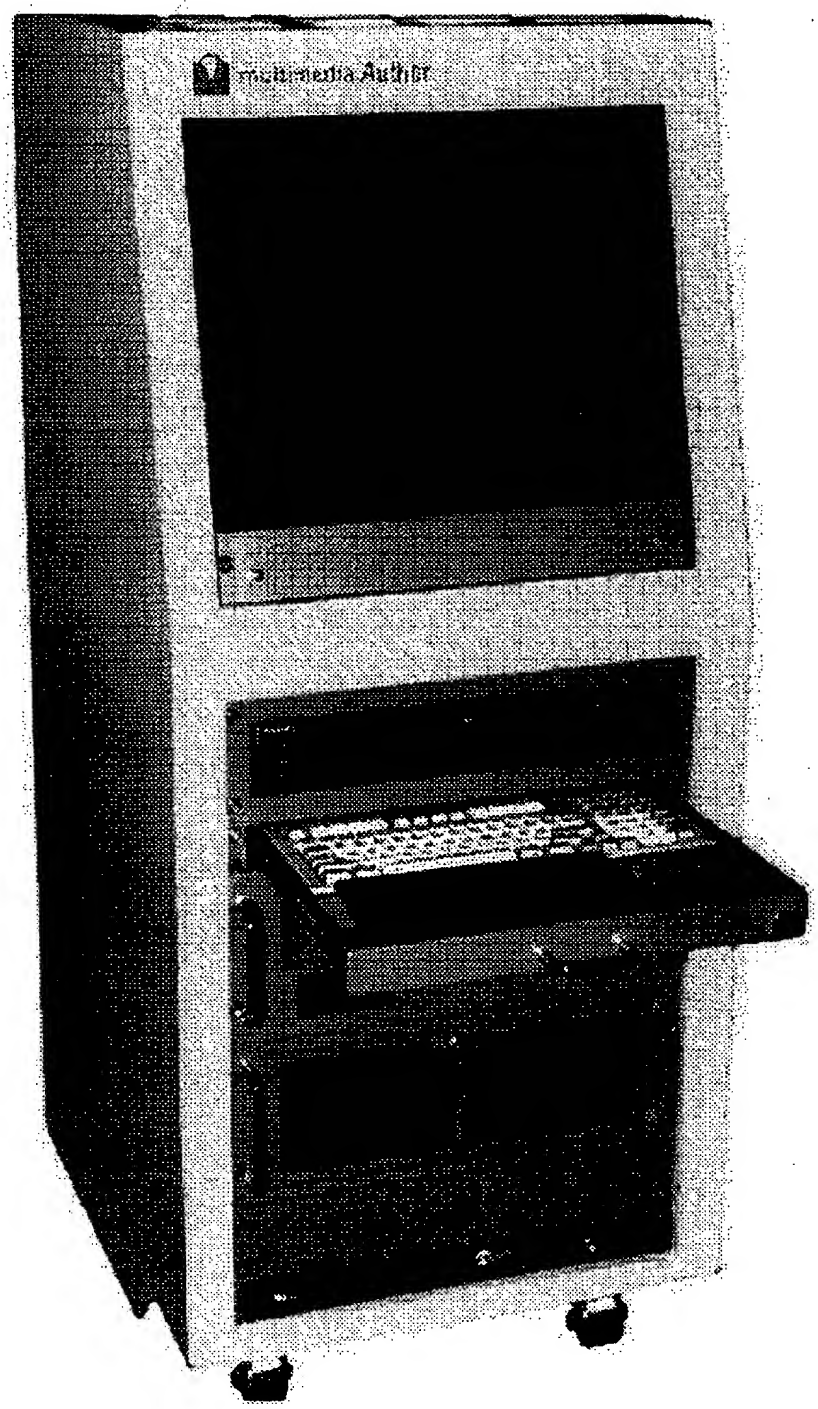


FIG. 7

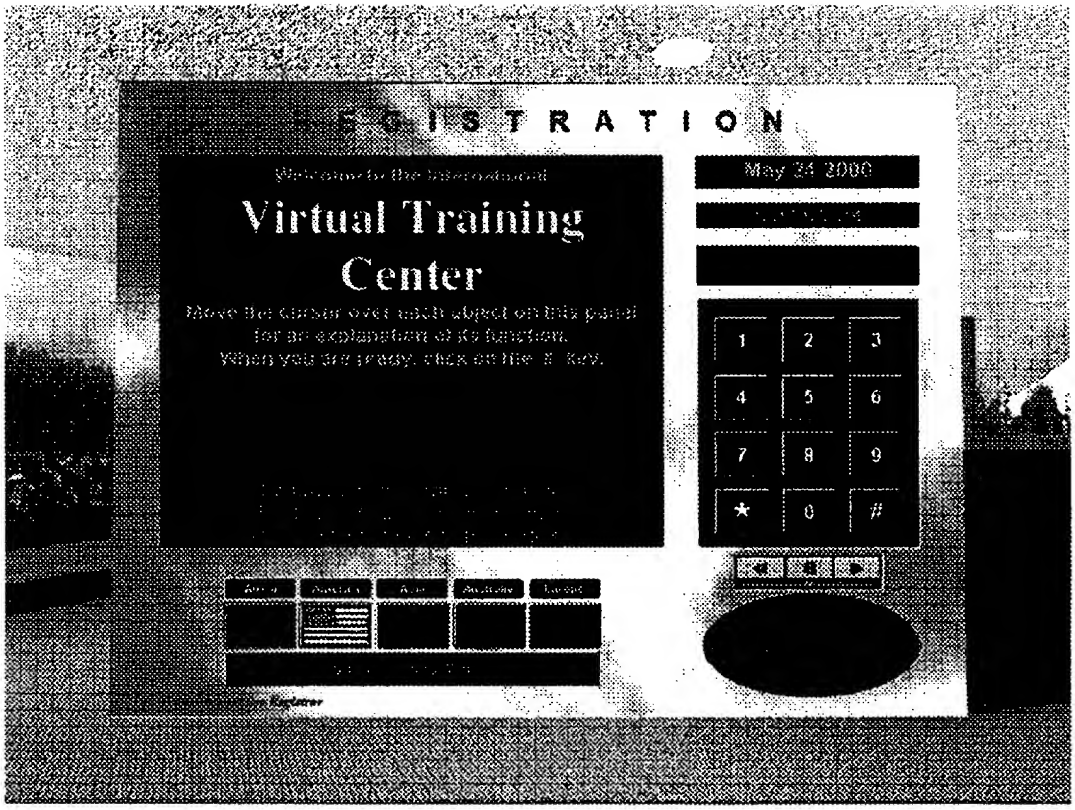

REGISTRATION
Virtual Training
Center
May 24 2000
1
2
3
4
5
6
7
8
9
*
0
#

FIG. 8

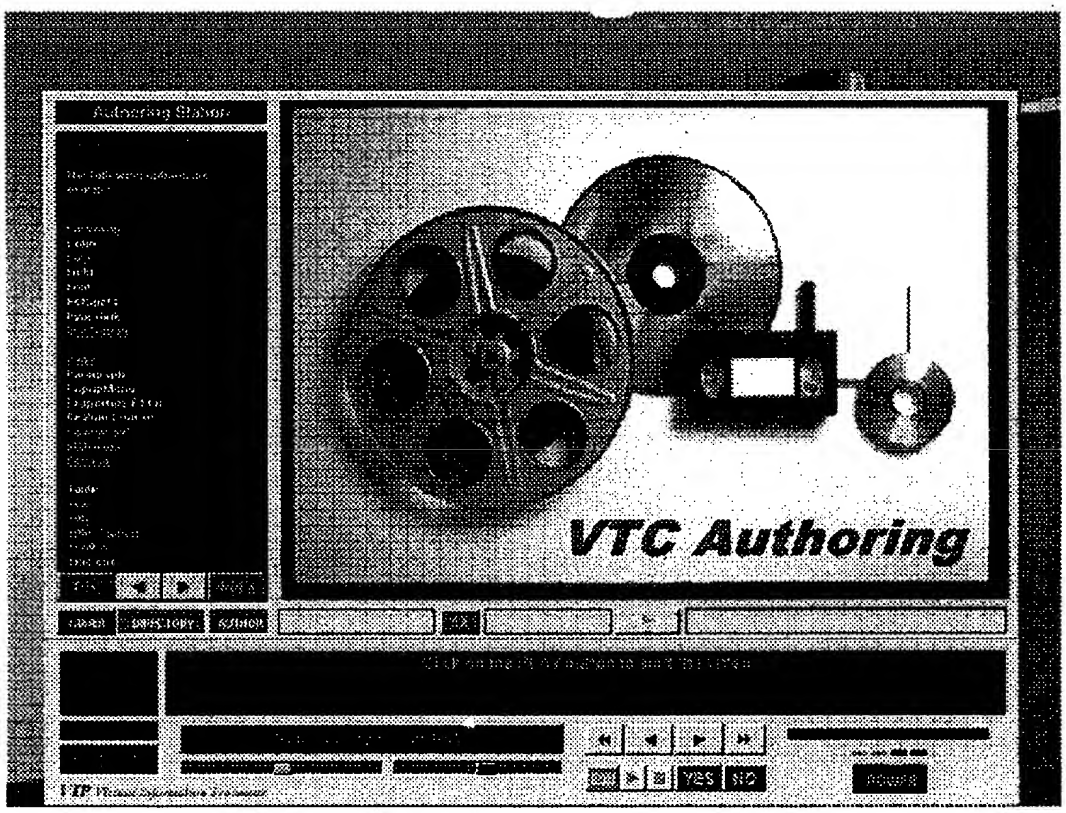


FIG. 9

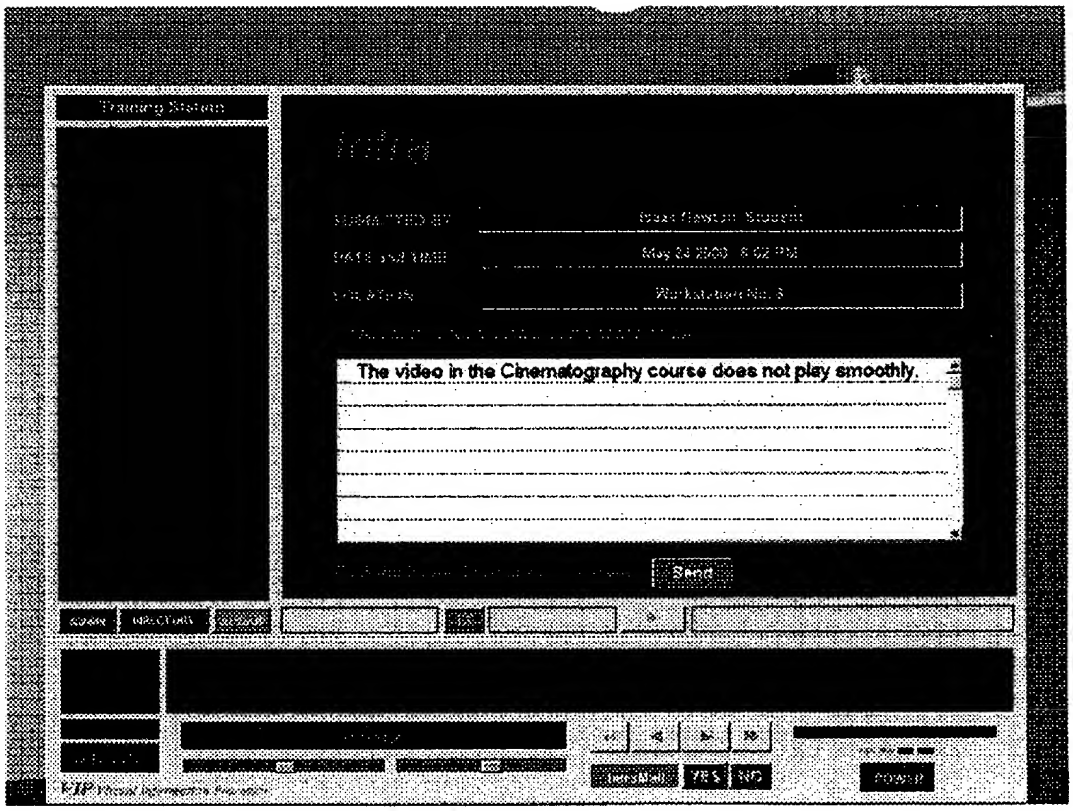

The video in the Cinematography course does not play smoothly.
Send
YES
NO

FIG. 10

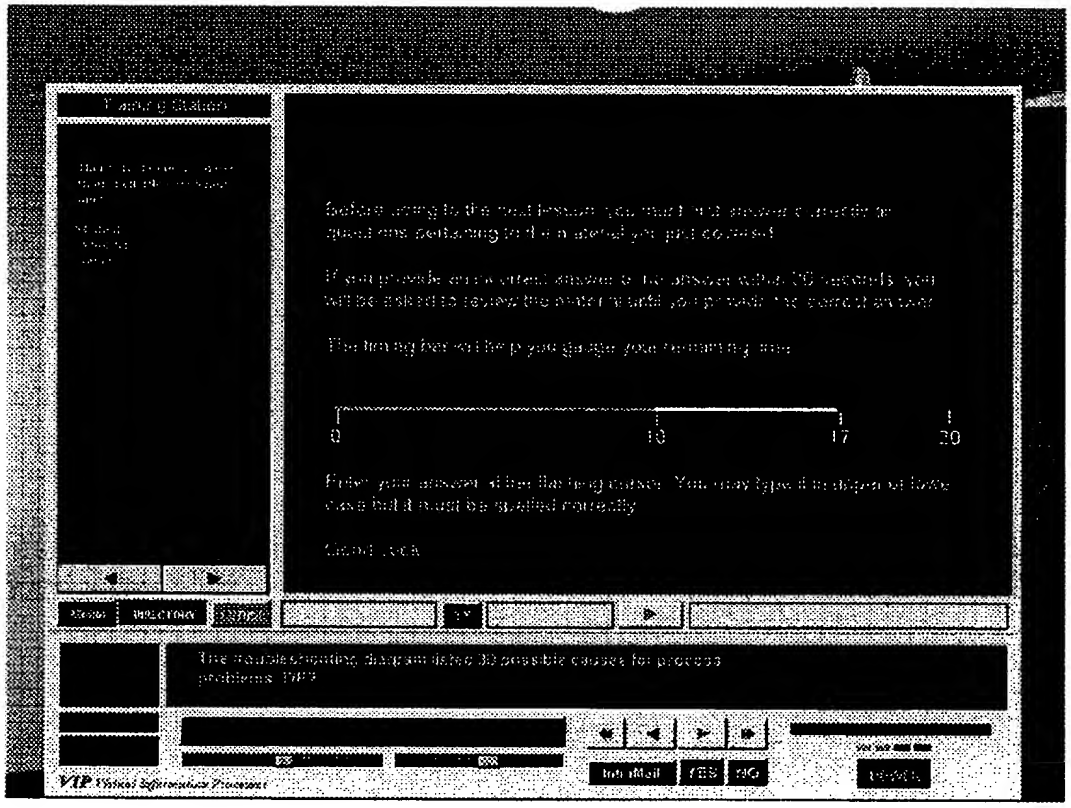


FIG. 11

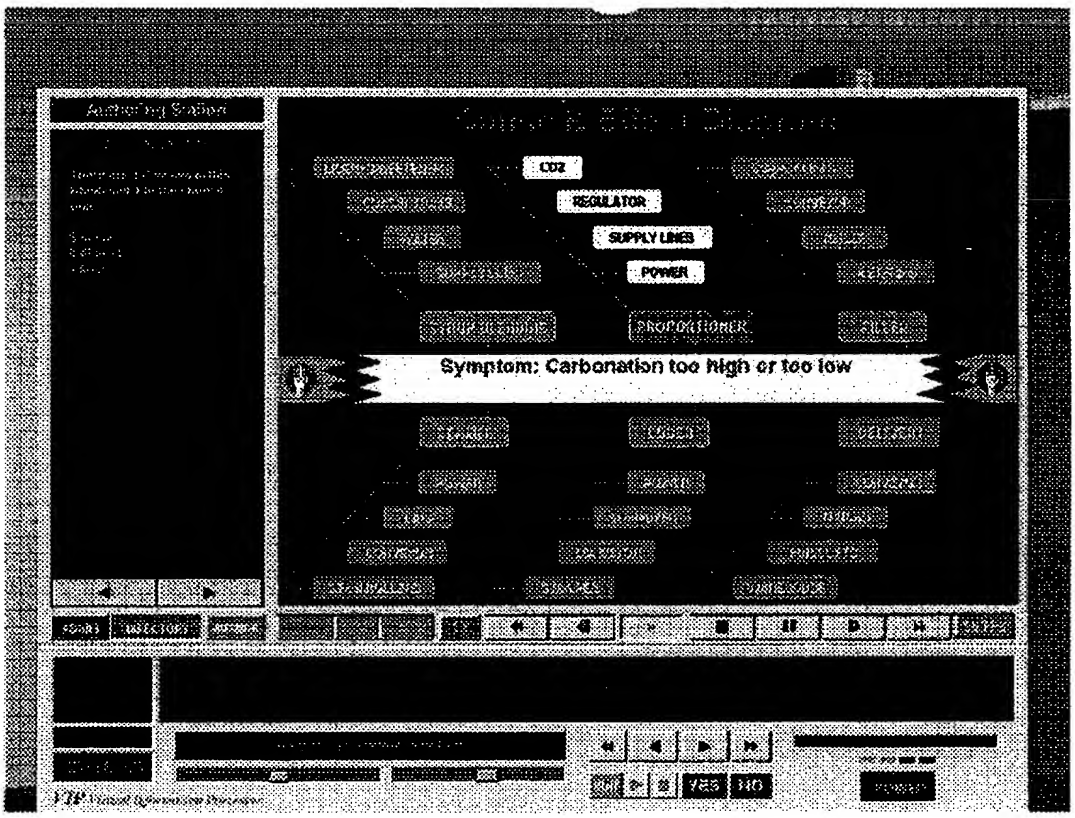

CO2
REGULATOR
SUPPLY LINES
POWER
Symptom: Carbonation too high or too low

FIG. 12

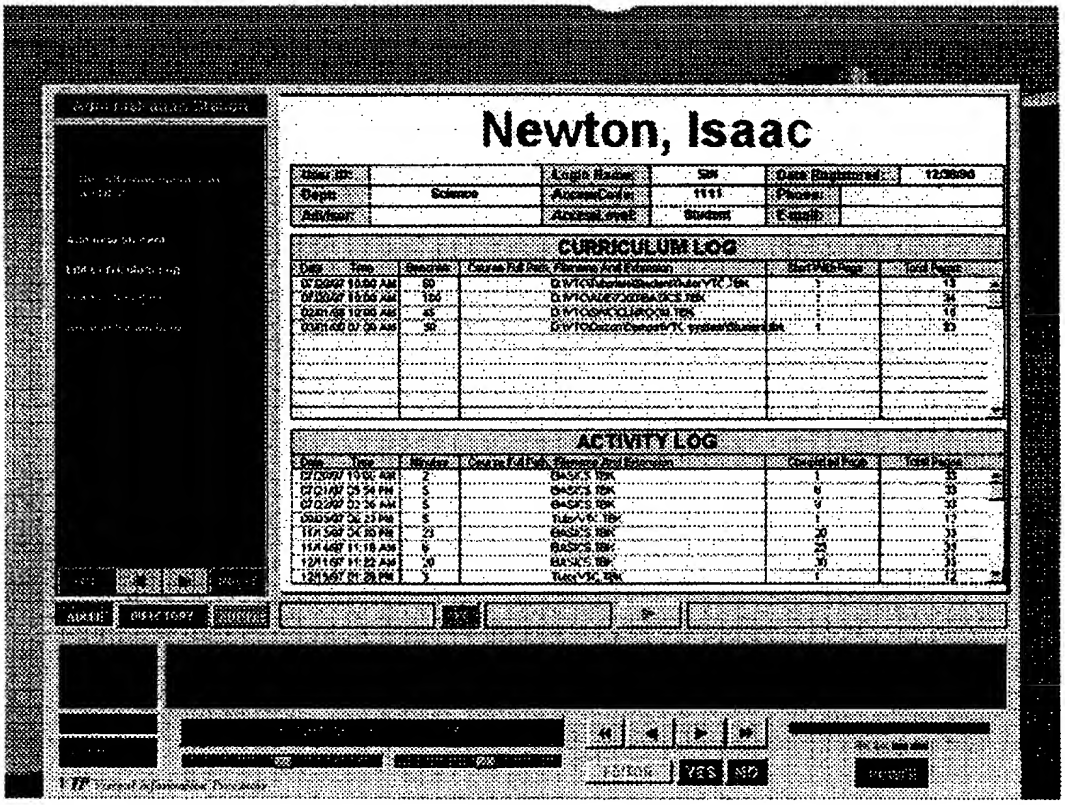


FIG. 13

FIG. 14

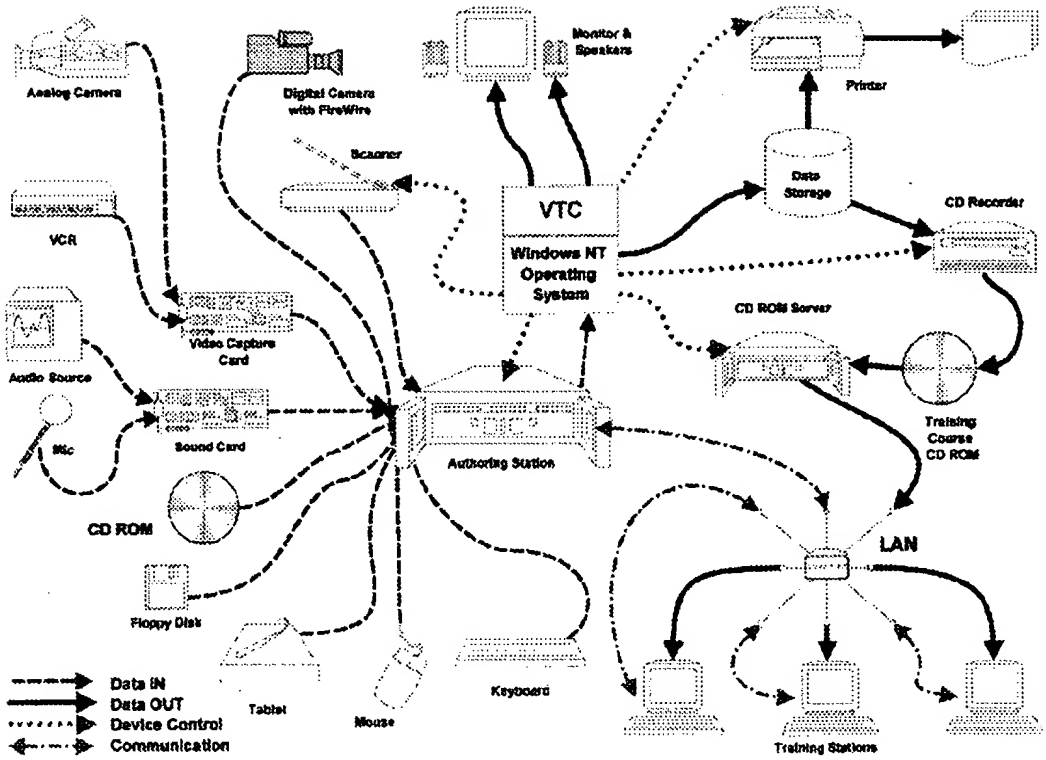

Aeslog Camera
Digital Camera with FireWire
Monitor & Speakers
Printer
Scanner
VCR
VTC
Windows NT Operating System
Data Storage
CD Recorder
Video Capture Card
CD ROM Server
Audio Source
Mic
Sound Card
Authoring Station
Training Course CD ROM
CD ROM
LAN
Floppy Disk
Tablet
Mouse
Keyboard
Training Stations
Data IN
Data OUT
Device Control
Communication

FIG. 15

AUTHORING SYSTEM AND METHOD FOR COMPUTER-BASED TRAINING

This application claims priority from U.S. Provisional patent application Ser. No. 60/052,792, filed Jul. 7, 1997.

FIELD OF THE INVENTION

The invention is related to the field of computer-based training and, in particular, to a system for authoring, administering, and deploying an interactive computer training curriculum.

BACKGROUND OF THE INVENTION

Computer-based training has been used successfully by industry for many years because it provides many advantages over conventional instructor-led training classes. Unlike a classroom in which the students and instructor must meet physically in the same location at the same time, computer-based training can be made available at any time and place that is convenient for individual students. Computer-based training developed at one site can be used around the world. Unlike classroom instruction, in which the instructor must target training speed to the majority of students, bypassing slow learners and boring fast learners, computer based training is one-to-one, so that each individual can proceed at his or her own pace.

Students retain more of what they learn in computer-based training because it is possible to reinforce the learning by having each individual student perform some task on the computer repeatedly to apply the newly acquired knowledge. Computer-based training also allows students to review their training skills as needed and when a refresher course is necessary, without the usual scheduling conflicts and the expense of an instructor. Current computer-based training systems suffer from several drawbacks. Computer-based training courses, particularly multimedia computer-based training courses, are very expensive to develop. One commonly used estimate for the cost of computer-based training is that each minute of training costs $1,000 to produce. This is based on approximately 2000 hours of work required to create a single hour of computer-based training at $30 per hour.

Producing a computer-based module requires expertise in the course subject matter as well as expertise in technical areas such as recording audio and video information, interfacing the recorded information with a computer, editing the audio and video information, page design, and educational theory. Merely figuring out how to digitize and input audio or visual data of a particular format into an editing program that requires a different format, can be a formidable task. Inferior computer-based training modules that are ineffective result from computer-based training modules being produced by individuals that lack any of the many skills required to produce a good training module.

Another problem with computer based training is the discomfort that many students feel with computers. Students are required to learn not only the subject matter, but also how to operate the computer and several applications with different look and feel, which can cause stress that reduces the effectiveness of learning.

SUMMARY OF THE INVENTION

An object of the invention is, therefore, to provide a training system that conveys a maximum knowledge to the student for a minimum of resources.

Another object of the invention is to provide a non-intimidating learning environment that minimizes the student's attention on the training tools, that is, the computer hardware and operating software, to maximize the student's attention on the subject matter being taught, to encourage the students and to ensure they can perform the trained tasks with ease.

Yet another object of the invention is to remove technical barriers to the authoring of computer-based training so that authors are not required to become computer engineers or programmers.

Still another object of this invention is to provide training system administration that ensures students follow a prescribed curriculum and that faithfully records training activity.

Yet another object of the invention is to provide an integrated training system that provides standardized, compatible, extensible tools for authoring, learning, and administering training.

The present invention is an integrated system that includes functions for authoring, operating, and maintaining a computer-based training program. In one aspect of the invention, the system includes integrated hardware and software that allows an author to create computer-based training modules that include audio, video, and textual content without requiring any programming. The author is not required to be aware of the underlying operating system of the computer or of the details of interfacing between the authoring computer and the recording tools. The system is extensible in that the author, if so desires, can access the underlying operating system and programming tools to create training content that is beyond the capability of the original system. The system also allows the integration of other technologies, such as audio, video and imaging formats or other media, without requiring the author to perform hardware or software interfacing.

Another aspect of the invention is the use of a training center metaphor that serves as the basis of the interface between the users and the system. Although the subject matter is presented on a computer terminal screen, images on the screen create the impression of a physical training center. Cues that indicate that a computer is being used are minimized or eliminated. The virtual training center provides a nonintimidating visual environment that is open and relaxing, thereby preparing the student to learn rapidly.

To create a non-intimidating environment and to fix more of the student's attention on the subject matter to be learned, rather than on the learning tools themselves, the interface maintains as much as possible the illusion of being in a real training center. To this end, manifestations of the operating system, such as computer icons and pop-up menus, are minimized or eliminated. By minimizing computer symbols, a student's discomfort with the computer can be reduced, and the student can concentrate on learning the material, not on learning how to operate and interact with the computer. All learning modules created with the invention have a similar look and feel so that the user can become comfortable with a single interface that is used for training, authoring, and administering varied training courses and programs.

To conform to the metaphor of an open, relaxing learning environment, the virtual training center may display, for example, an open room with training terminals partially walled off from each other to the extent that would be required to reduce distractions for real students. A large portion of the outside walls of the virtual training center may

be windowed, showing a quiet, relaxing scene, such as a serene mountain lake or a desert. The lack of computer symbols, such as icons and pull down menus serves to maintain the illusion of a real, physical training center: Real physical training centers do not have pull down menus.

The system also performs administrative functions, such as ensuring that students take courses in proper sequence, tracking training activity and course completion, and notifying the course administrator when a course is completed.

The subject matter of the present invention is particularly pointed out and distinctly claimed in the concluding portion of this specification. However, both the organization and method of operation, together with further advantages and objects thereof, may best be understood by reference to the following description taken in connection with accompanying drawings wherein like reference characters refer to like elements.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** is a screen display showing an image of a student at the Registration Board at the entrance of the Virtual Training Center;

FIG. **2** is a screen display showing an image of a student having entered the Virtual Training Center of FIG. **1**;

FIG. **3** is a screen display showing an image a workstation in the Virtual Training Center of FIG. **1** from the viewpoint of a student approaching the workstation;

FIG. **4** is a screen display showing an image of two Virtual Interactive Players from the viewpoint of a student entering the workstation in the Virtual Training Center of FIG. **3**;

FIG. **5** is a screen display showing an image from the viewpoint of a student of leaving the Virtual Training Center of FIG. **1**;

FIG. **6** is a screen display showing an image from the viewpoint of a student having left the Virtual Training Center of FIG. **1**;

FIG. **7** shows a typical workstation used with the present invention;

FIG. **8** is a screen display showing the Registration Board of FIG. **1**;

FIG. **9** is a screen display showing on the Virtual Interactive Player of FIG. **4** a typical author's Virtual Interactive Player Directory

FIG. **10** is a screen display showing on the Virtual Interactive Player of FIG. **4** a Request Form for communicating between system users;

FIG. **11** is a screen display showing on the Virtual Interactive Player of FIG. **4** a typical review question presentation;

FIG. **12** is a screen display on the Virtual Interactive Player of FIG. **4** showing a typical fishbone diagram;

FIG. **13** is a screen display a showing on the Virtual Interactive Player of FIG. **4** a table of contents for administrators;

FIG. **14** is a screen display showing on the Virtual Interactive Player a typical student's Virtual Interactive Player Directory; and

FIG. **15** shows the interrelationship of the hardware components of the present invention.

DETAILED DESCRIPTION

There are three user levels for logging onto the system: author, administrator, and student. The invention is most clearly explained by describing the aspects common for all users and then the separate but integrated functions of the three different types of users.

Common Aspects

The preferred embodiment uses a metaphor of a training center to create a consistent, non-intimidating, relaxed learning environment. FIGS. **1–6** show an example of how a Virtual Training Center (VTC) would appear to a user. The non-intimidating environment of the virtual training center metaphor is maintained for all three levels of users.

The invention is preferably implemented using an industrial computer workstation with a large screen, high resolution monitor. (FIG. **7**) The display presents the training material on a portion of the computer terminal screen and shows the background of the virtual training center on the remainder of the screen. (FIGS. **8–14**) For example, in a monitor that supports a resolution of 1024×768 pixels, the learning material is presented in a 640×480 pixel area and the remainder of the monitor displays a background showing a virtual training center. The constant display of the virtual training center in the background keeps the user psychologically connected to the virtual training center and helps prevent disorientation.

Controls on the screen are designed to resemble mechanical devices. For example, the student plays a video clip of an operation by using a control that resembles a control on a familiar video cassette player. Sound effects, such as motors turning and other mechanical sounds, are used to simulate the mechanical device to enhance the perception of operating a real mechanical device. The function of each mechanical control, such as knobs and pushbuttons, is displayed on the device's display screen as a mouse cursor approaches a control.

As a user begins a session, the monitor displays an image of a Registration Board (FIGS. **1** and **8**) at the entrance of the Virtual Training Center. The system identifies the access level of a user as a student, author, or administrator as they log in using their assigned identification and password. The Registration Board can also provide a choice of languages for the training. The user is instructed, verbally and in text, in the log-in procedure in several languages. Selecting a particular language, can cue the Application to present the course in that particular language, provided it has been authored as such. If the user enters a valid identification and password, the display shows the perspective of an individual moving towards an entry door. The user then presses a button to enter the Virtual Training Center. By pressing the button for entry, the user is performing an affirmative act to psychologically prepare himself for authoring, training or administration.

Upon entering the training area of the cylindrical Virtual Training Center, the Application shows a student's perspective moving towards an Atrium at its center. The Virtual Training Center is shown in a 360° panoramic view, having exterior windows that look out upon a serene vista, such as a mountain or a desert scene. Even during actual training, a portion of the screen maintains the image of the Virtual Training Center, thereby maintaining the student in a state of mind conducive to training by having him feel he is in a quiet, peaceful setting, as opposed to a noisy workplace. The user is then guided through the Atrium and proceeds to one of several training workstations, that is, a "Multimedia Instructor" containing a Virtual Interactive Player ("VIP").

At the VIP, the user is presented with the VIP Directory that shows the options accessible to the user. The options

will depend on the type of user and on the individual student's curriculum. The VIP Directory is essentially a menu appearing in the form of a 5-part dynamic directory (FIG. 9). It indicates the options that are available to the user or that will be available to the user in the future, such as courses that are part of the student's curriculum but are unavailable until prerequisites are completed. The options are color coded: green options execute when they are selected and white options open another set of sub-options. The directory sub-options can go up to four deep.

The selections run different applications. Each application has a table of contents. Maintaining the metaphor of a physical environment, the table of contents of each application appears as it would in a real book. The application also includes an alphabetical list of topics. Both the Table of Contents and the List of Topics are generated and updated a automatically as the author assembles or edits a course.

A common item to all users is a Request Form (FIG. 10), that is, an internal messaging system by which the students, administrators, and authors can communicate about training issues. The messaging system is not interconnected with any other network, so that only training related messages are passed through the system, thereby removing non-training related distractions during training sessions.

To be consistent with the virtual environment, when a user is ready to end the current session, he simply "pushes" to POWER button to turn the VIP off. The Application then shows the visual perspective of the user as he backs off the workstation, turns towards the exit doors of the VTC which open automatically letting external sounds of nature such as bird songs to be heard. (FIGS. 5 and 6)

Authors

To help an author design and organize a course before beginning to write and videotape substantive material, the System makes available to the author an Instructional Design pre-authoring tool, such as Designer's Edge from Allen Communications, Salt Lake City, Utah.

After designing and organizing a course, the author begins to construct it by assembling the various media. To create a complex instruction for a particular operation, such as the calibration of a measuring instrument, an author may choose to videotape the procedure instead of describing it in text and illustrations. The video is then digitized and stored into the computer. The author can divide the video into portion, which can be as short as a single frame. Controls on the display let the author move through the video frames forward or backward to determine exactly which frames should be included in any individual step of the training.

Portions of the video are assigned step numbers and the author can associate audio information and text with each step. Information is preferably passed to the student visually or aurally, as opposed to textually, to improve retention. Text can be associated with different areas of a displayed image so that when the student moves the cursor over that portion of the image, the text is displayed. For example, when a machine is displayed, the display can show the names of parts of the machine as the cursor is moved over those parts. By not displaying information until the cursor is over the region of interest, the display is less cluttered and conveys information more clearly to the students.

Various known video compression formats can be used. For example, the MPEG standard using 3× real time compression provides the advantage of near real time data compression with the flexibility of being able to edit the digitized material. Other video capture and compression standards, such as the Intel Indeo, can also be used although they typically do not support near real time compression. When a digitized video file is edited to produce video portions for a training module, the original file is saved so that it can be called and edited again if necessary.

The author has the ability to place tests at different points of the course. For example, the author may place tests after every screen to test that the student has learned the information presented on each screen. The tests are typically made up of questions that invite a student response that requires a minimum amount of typing, such as true or false questions or questions that require a number for the answer. Such questions can be easily interpreted by the computer. The student will have a time limit, typically 20 seconds, for answering each question on a test. After a question is posed, both an analog and a digital timer are displayed, showing the time remaining for answering the question and changing from green to yellow to red as the time expires. If the student does not pass the test within the allowed time, the screen containing the material that is being reviewed is redisplayed and the student has another opportunity to learn the information. The student is sent back after failing a test question as often as necessary until the student learns the information and can answer all questions correctly.

After each section of the training material, the author can place an overview of what was presented in that section. (FIG. 11) There can also be tests after a section of material is presented. If the student answers any of the questions incorrectly on the section test, the screen re-displays the page at which the information required to answer the question is introduced. This provides the student another opportunity to learn the material, before returning to the section test.

Similarly, there can be an overview at the end of the entire course and then a test on the material presented in the entire course. Incorrect answers on the overall test will also cause the screen in which the unlearned information was presented to be displayed and then the student will return to the overall test. Students have the opportunity to repeat the training material as often as necessary to learn the subject matter. Because the student's responses are not recorded by the System, and incorrect answers are not observed by other students or by an instructor, the student will be more relaxed and more able to assimilate information.

Thus, the student is preferably questioned about the subject matter three times: immediately after it is learned, at the end of the section, and at the end of the training module. The student typically must answer all of the questions correctly to be passed on to the next level. The number of questions at each level of review typically increases. Students can go back in the course to review sections already completed but cannot go forward out sequence. Once the entire course is completed, the student can access the entire course in any order much like a textbook becomes a reference after the course is completed. Of course, an author can arrange the review questions as he feels appropriate.

One embodiment of the invention includes capabilities to allow an author to easily create interactive "fishbone" (Ishikawa) diagrams (FIG. 12). These type of diagrams are very effective in troubleshooting by visually depicting the relationships of cause and effect. All objects comprising the interactive diagram are pre-designed so that the author only needs to add symptoms and remedies to construct an effective troubleshooting diagram. The student is guided visually and systematically towards a solution. At the conclusion of a troubleshooting session, the date, time, user's name,

symptom and actions taken are recorded in the Troubleshooting Log database. The database can then be used to derive statistical information pertaining to a particular system such as frequency and type of failures by system or component.

An author can access screens of the program out of sequence, whereas a student must follow the prescribed order. An author is also allowed to access the underlying operating system for finding files or other administrative functions. One implementation of the invention uses the Window NT® operating system from Microsoft Corporation, Redmond, Wash. Other operating systems may also be used.

In the Author mode, the VIP has an additional button that the student never sees. When engaged, this button elevates the Editing Menu to full view. The menu includes several editing functions that the author can use to add, delete and modify the course content. All this is accomplished without programming. When disengaged, the EDIT button retracts the Editing Menu below the tabletop that supports the VIP and the Table of Contents as well as the Index of Topics are updated automatically.

One implementation of the inventive system was written primarily in ToolBook II from Asymetrix Corporation, Bellevue, Wash. This implementation includes both a runtime and a full version of ToolBook II. The full version of ToolBook II is only accessible by an author. The author then has the option of programming functions and enhancements beyond the capability of the VTC software system for incorporation into the training courses.

Administration

When an administrator logs on, the presented table of contents (FIG. 13) includes selections that allow the administrator to review and maintain student records, generate reports of such things as user activity and VTC activity, and access utilities, including the master course list, log-in names and passwords, re-certification schedules and issuance of certificates of completion.

The system includes a record for each student. The student record includes the date the student registered, the student activity log, and the student curriculum. The student records are stored on a separate computer so they cannot be edited by anyone except the administrator. To ensure integrity of the training records, the user activity information cannot be edited by anyone, including the administrator. The data can therefore be relied upon for certifying training for customer audits and to qualify for international certifications, such as ISO 9000.

The administrator works together with the student to define an appropriate curriculum for the student.

Various jobs require re-certification at regular time intervals. The system can generate a list of students requiring re-certification and manually or automatically, notify the students promptly.

The administrator maintains a Master Course List of all the courses available. The VIP Directory is generated automatically from this List.

Statistical information and graphs on the number of courses available over time, the average amount of time required to complete a course, the actual number of students taking and completing courses and other useful statistics are generated automatically from the student records. This information can be used by management to evaluate Return On Investment, help with budgeting and justify additional funding of the training program.

Students

After the student enters the Virtual Training Center and proceeds to a training workstation, the terminal displays his curriculum on the VIP Directory. (FIG. 14) It includes courses that the student has completed and courses to be completed. The student can only enter a course for which he has completed the prerequisites. A student can access courses only in the sequence specified by the administrator. As soon as a prerequisite course is completed, the next course becomes available for access automatically. Students are permitted to go back over completed courses for review or reference.

The VIP Directory begins with a general level and expands to specifics as the student makes selections. For example, the top level may have a class of particular machines for which training modules exist. The second level may include machines of a certain type within this class, the third level may include different manufacturers of this type of machines, the fourth level may include specific models systems of the selected machine and the fifth level may include training courses on basic theory of operation, operation, maintenance, etc. Before the student can select any particular module, he checks on the Directory to see whether he can take that course.

Once a course is selected, a mechanical sound is heard resembling the sound of a real CD ROM being inserted into a drive and the CD slot of the VIP is shown to be loaded with a virtual CD ROM that carries the course title. This feature is desirable not only to maintain the illusion of operating a real machine but to also indicate to the student the subject matter he is studying at all times. The CD ROM insertion is followed by the opening of the course's title page and from this point on the student is in control of the courses pace. Page navigation is accomplished with Forward and Reverse buttons.

The VIP's digital clock is a multifunction device that automatically displays the time of the day, elapsed seconds during a review or the frame number of a digital video. The student can also use it as a stopwatch by just clicking on it with the mouse.

The virtual monitor of the workstation has seven built-in pushbuttons that the student can use to fully control the digital video material included in a course. Thus a student can play, stop, pause, repeat, fast forward or single step forward or back any portion of a video procedure that needs to be examined or studied carefully. This level of control is very important in the learning process yet it is usually found only on very expensive professional video editing equipment.

When a student completes a course, notification of the completion is sent to the administrator automatically. The curriculum is then updated either manually by the administrator or automatically. After the curriculum is updated, the VIP Directory will indicate to the student that he can take courses for which the completed course is a prerequisite. Although automatic updating is easier, manual updating ensures that the administrator is aware of the progress that the students are making. Upon completion of a course, the system can print a completion certificate for presentation to the student by the training administrator, the student's supervisor, or other appropriate personnel. Also upon completion of a course, the student is allowed to access the material for that course in any order, rather than merely sequentially. The student can then use the course as a reference in his day-to-day performance of the task taught by the course.

To close the current course, a student merely presses the DIRECTORY button which retracts the CD ROM out of sight and returns to the VIP Directory.

Workstation

In a preferred embodiment, the system comprises the following hardware and software. FIG. 15 shows the relationship between the hardware components of the system. The system is not limited to the hardware and software below, which are listed only as examples.

System Software

High performance Operating System (e.g., Microsoft Windows NT 4.0 Workstation)
Hard Disk Defragmenter
Authoring, administration and deployment software (Orizon Multimedia VTC 97)

Application Software

Authoring (e.g., Asymetrix ToolBook II)
Instructional Design (Allen Communications Designer's Edge)
Word processing, Spreadsheet, Database (Microsoft Office Professional)
Vector/bitmap graphics, Flowchart, Animation (Graphics Suite)
Non-Linear video editor
Rae Sound wave editor
OCR software
Image capture and editing

Computer System

High performance CPU (Intel 200 MHz Dual PentiumPro)
Adequate Random Access Memory (128 MB or more)
A/V rated HDD
Ultra Wide SCSI controller
3.5" FDD
4× CD Recorder
SCSI CD ROM drives
Full function Keyboard
Pointing device
High speed LAN adapter

Video/Audio System

1024x768 color monitor
True Color (24 bit) video accelerator
Editable, real time video capture at 30 fps near real time MPEG-1 compression
Hi-8 VCR
Hi-8 Camcorder
Flatbed color scanner
16-bit sound card
Stereo headset with boom mike

Console

19" rack
CD ROM drive bay
Drawer for keyboard and pointing device
Drawer for scanner
Power: 110 VAC

An embodiment of the present invention is suitable for use as a training standard, providing the advantage of allowing students to use a standard training setting for training on a variety of topics, thereby reducing the requirement that students learn the use of different training tools and allowing the student to concentrate on the subject matter, rather than on the training equipment.

While a preferred embodiment of the present invention has been shown and described, it will be apparent to those skilled in the art that many changes and modifications may be made without departing from the invention in its broader aspects. The appended claims are therefore intended to cover all such changes and modifications as fall within the true spirit and scope of the invention.

What is claimed is:

1. A method of self-paced, computer-based training for individuals, the method usable in a distracting work environment by focusing the individual's attention onto the training, comprising:

providing the student with a computer screen and an input device;

providing on the computer screen a series of images of a student entering a virtual training center and moving through the virtual training center toward a virtual training workstation, the images of entering and moving through the virtual training providing a virtual environment to focus the student away from the real environment and onto the training;

providing on the computer screen an image of the virtual training workstation with the virtual training center in the background, the virtual training center image in the background providing the student a virtual training environment, thereby reducing distractions from the outside, non-virtual environment;

presenting to the student on the virtual training workstation a choice of training programs for presentation, the training programs presented including only those programs that the student is qualified to undertake;

presenting to the student on the virtual training workstation an audio-visual presentation including first information to be learned;

presenting to the student on the virtual training workstation a test to determine whether the student has learned the first information;

if the student has learned the first information, presenting to the student on the virtual training workstation second information; and

if the student has not learned the first information, repeating the presentation of the first information.

2. The method of claim 1, further comprising automatically sending a message to a system administrator when the student learns a pre-defined set of information.

3. The method of claim 1, in which providing on the computer screen an image of the virtual training workstation including providing lesson control features to the student, the lesson control features being presented as images of physical control devices, rather than as conventional computer control imagery.

4. In a method of computer-based training in which a student is presented information at a student-controlled pace and in which the student is tested periodically and re-presented information that the testing shows was not learned, the improvement comprising providing the student a virtual training center environment metaphor to symbolically provide the student a virtual training environment separate from the working environment.

5. The method of claim 4, in which providing the student a virtual training center environment includes presenting the student with an environment that simulates a physical, non-computer training environment.

6. The method of claim 4, in which providing the student a virtual training center environment includes presenting the student with a portion of a computer screen that represents a physical training environment.

7. The method of claim 4, in which providing the student a virtual training center environment includes showing a virtual student entering a virtual training facility.

8. The method of claim 5, in which the virtual training environment includes an isolated, serene environment.

9. The method of claim 4, in which providing the student a virtual training center environment metaphor includes providing the student with lesson controls that simulate controls on physical devices, rather than computer-based controls, to minimizes computer-related imagery that can make students uncomfortable.

10. In a computer-based training apparatus that presents information to a student at a student-controlled pace and that periodically tests the student and represents information that the testing shows was not learned, the improvement comprising:

means for providing the student a virtual training center environment metaphor to symbolically provide the student a virtual training environment separate from the working environment; and

means for providing the student with lesson controls that simulate controls on physical devices, rather than computer-based controls, to minimizes computer-related imagery that can make students uncomfortable.

11. The apparatus of claim 10, in which the means for providing the student a virtual training center environment metaphor includes means for providing the student with lesson controls that simulate controls on physical devices, rather than computer-based controls, to minimizes computer-related imagery that can make students uncomfortable.

* * * * *

US006599130B2

(12) United States Patent
Moehrle

(10) Patent No.: US 6,599,130 B2
(45) Date of Patent: Jul. 29, 2003

(54) ITERATIVE VIDEO TEACHING AID WITH RECORDABLE COMMENTARY AND INDEXING

(75) Inventor: Armin Moehrle, Chicago, IL (US)

(73) Assignee: Illinois Institute of Technology, Chicago, IL (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 128 days.

(21) Appl. No.: 09/775,463

(22) Filed: Feb. 2, 2001

(65) Prior Publication Data

US 2002/0106623 A1 Aug. 8, 2002

(51) Int. Cl.[7] G09B 11/00

(52) U.S. Cl. ..... 434/365; 434/219; 434/307 R

(58) Field of Search ..... 707/512, 300; 434/262, 322, 365, 307 R; 600/217, 345

(56) References Cited

U.S. PATENT DOCUMENTS

4,416,293	A		11/1983	Anderson et al.
4,818,938	A		4/1989	Sattin et al.
4,945,410	A		7/1990	Walling
4,974,178	A		11/1990	Izeki et al.
5,208,748	A		5/1993	Flores et al.
5,235,510	A		8/1993	Yamada et al.
5,303,148	A		4/1994	Mattson et al.
5,544,649	A		8/1996	David et al.
5,583,980	A	*	12/1996	Anderson 345/473
5,619,995	A		4/1997	Lobodzinski
5,815,196	A		9/1998	Alshawi
5,836,877	A		11/1998	Zavislan
5,917,929	A		6/1999	Marshall et al.
5,920,317	A	*	7/1999	McDonald 345/835
5,924,069	A		7/1999	Kowalkowski et al.
5,978,014	A		11/1999	Martin et al.
5,982,352	A		11/1999	Pryor
6,014,451	A		1/2000	Berry et al.
6,049,769	A		4/2000	Holmes et al.
6,230,172	B1	*	5/2001	Purnaveja et al. 707/500
6,332,144	B1	*	12/2001	deVries et al. 707/102
6,449,653	B2	*	9/2002	Klemets et al. 348/423.1
6,452,615	B1	*	9/2002	Chiu et al. 345/716
2002/0059342	A1	*	5/2002	Gupta et al. 707/512

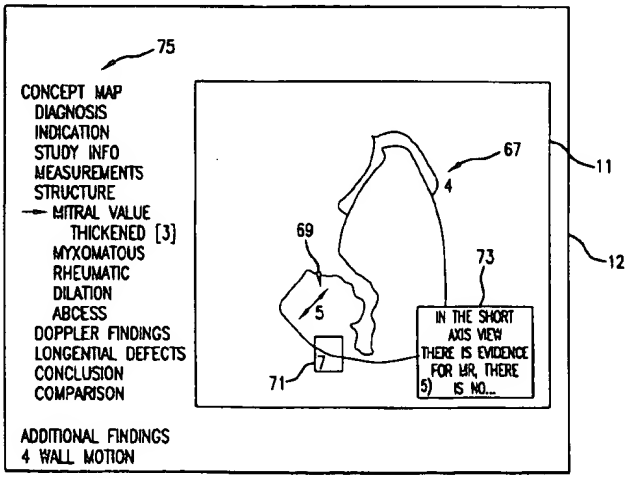
* cited by examiner

Primary Examiner—Joe H. Cheng
Assistant Examiner—Kathleen M Christman
(74) *Attorney, Agent, or Firm*—Pauley Petersen Kinne & Erickson

(57) ABSTRACT

A tutorial or diagnostic aid based on the representation and iterative interpretation of visual images is taught. A teaching or diagnostic session is created by overlaying an interpretation layer via a software application onto the visual image layer and synchronizing the two with the time code of the visual image. The interpretation layer allows the reviewer to identify image areas of interest by gesture and append comments thereto in real time; i.e. images or portions of images within the visual representation playback may be identified and labeled and have the concurrent commentary associated therewith. The comments are indexed and linked to a database of similar topics. The flow of the session is recorded to show the images, deictic gestures associated therewith, and commentary associated with the gestures, to enable subsequent users to playback a session and follow the flow of thought (i.e. image identification and commentary within the original session). Iterative sessions allow additional image identification and commentary to be accomplished. Additional database access through the index may further enhance a teaching session, or provide for research sessions and report generation.

25 Claims, 5 Drawing Sheets



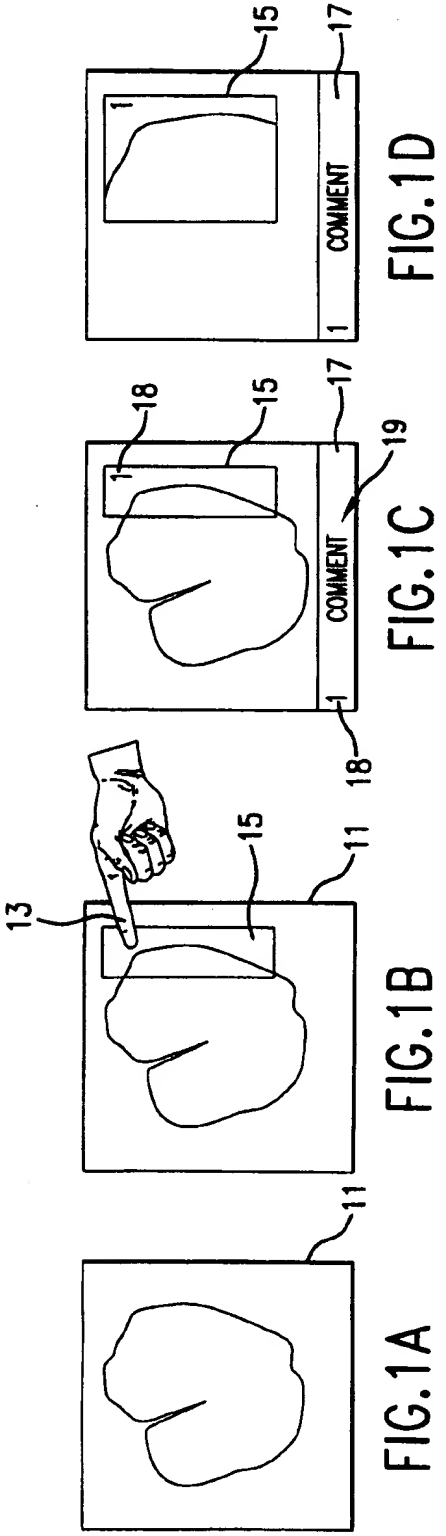


FIG.1A FIG.1B FIG.1C FIG.1D

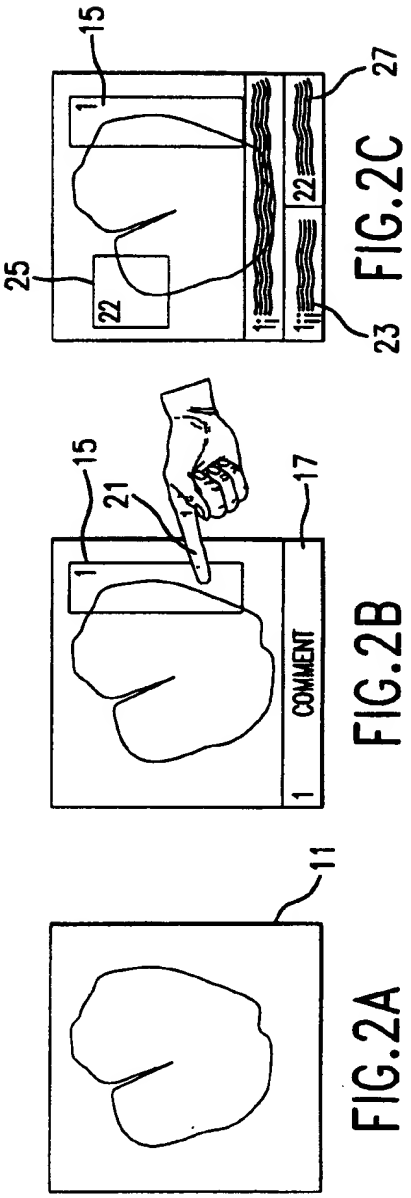


FIG.2A FIG.2B FIG.2C

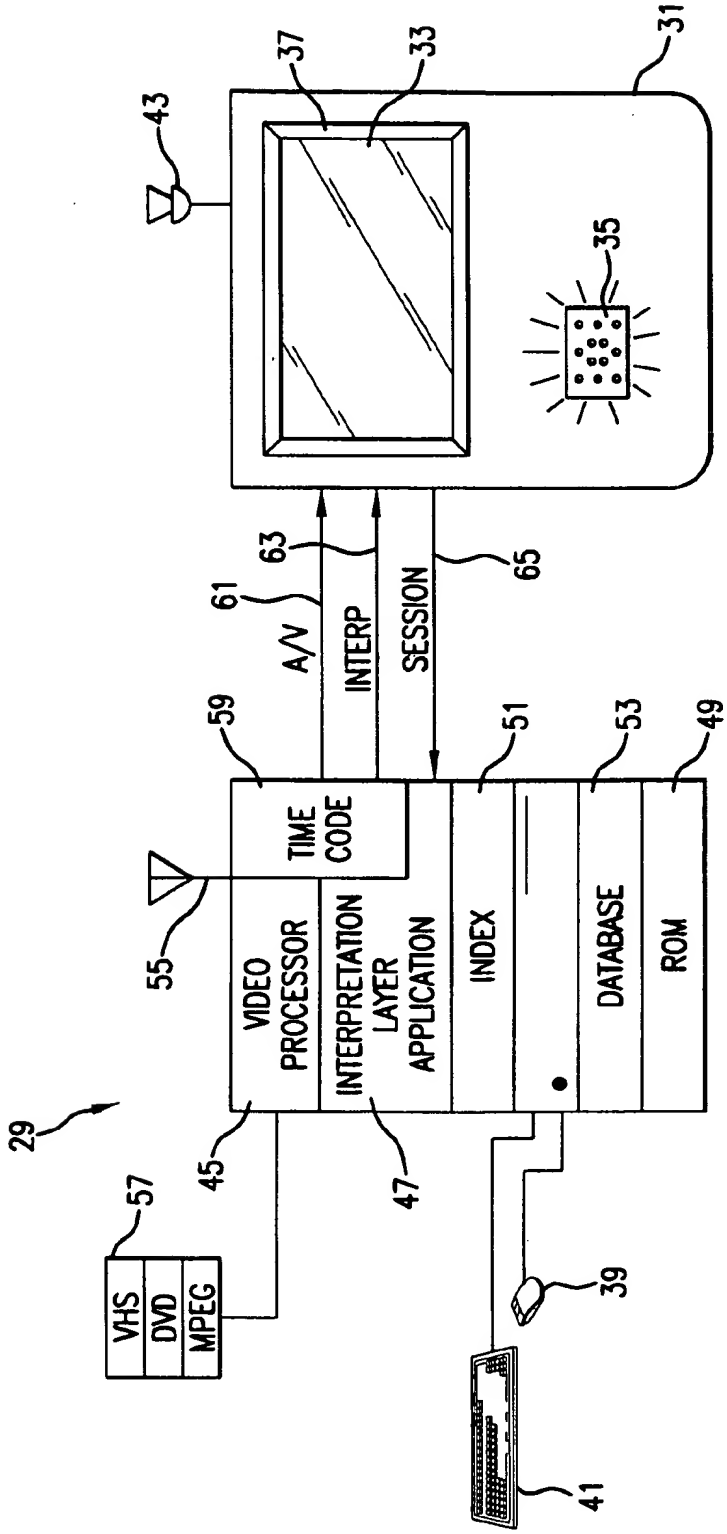

31
37
33
43
35
63
65
61
A/V
INTERP
SESSION
59
51
53
49
TIME CODE
55
VIDEO PROCESSOR
INTERPRETATION LAYER APPLICATION
INDEX
DATABASE
ROM
29
45
47
57
VHS
DVD
MPEG
39
41

FIG.3

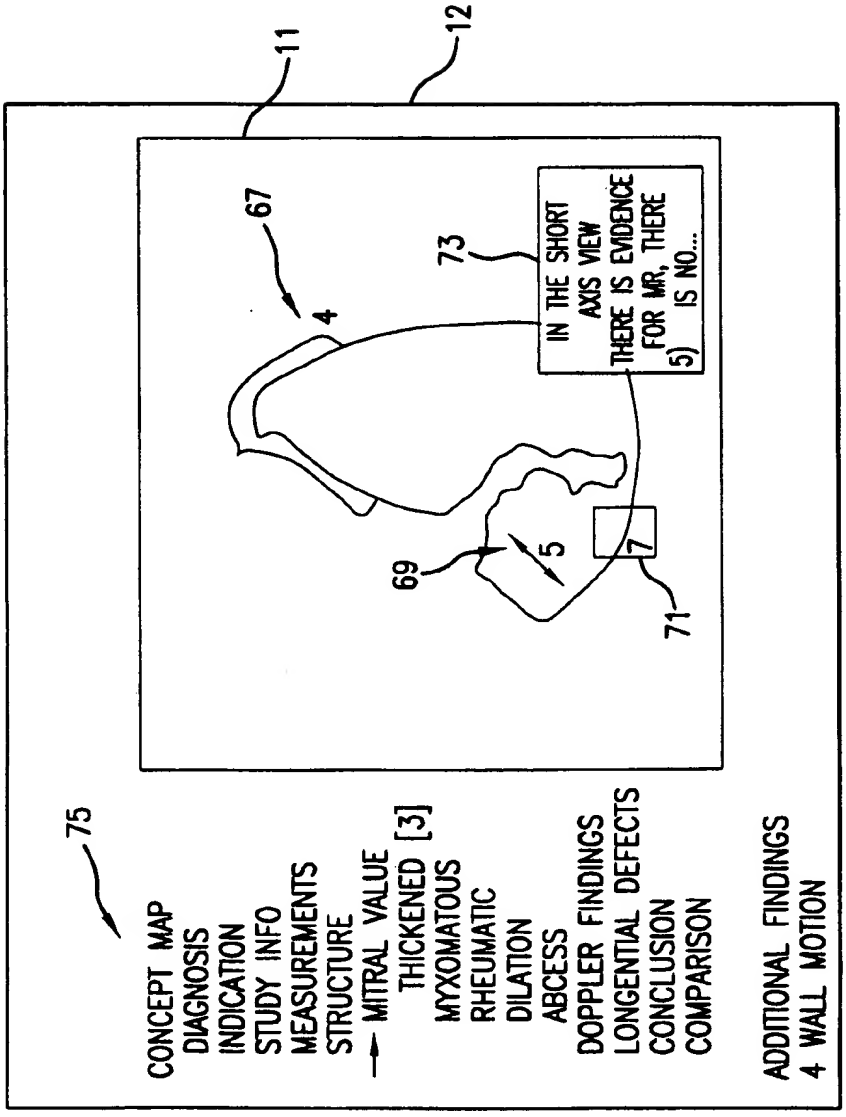

11
12
67
4
73
IN THE SHORT AXIS VIEW THERE IS EVIDENCE FOR MR, THERE 5) IS NO...
69
5
7
71
75
CONCEPT MAP
DIAGNOSIS
INDICATION
STUDY INFO
MEASUREMENTS
STRUCTURE
MITRAL VALUE
THICKENED [3]
MYXOMATOUS
RHEUMATIC
DILATION
ABCESS
DOPPLER FINDINGS
LONGENTIAL DEFECTS
CONCLUSION
COMPARISON
ADDITIONAL FINDINGS
4 WALL MOTION

FIG.4

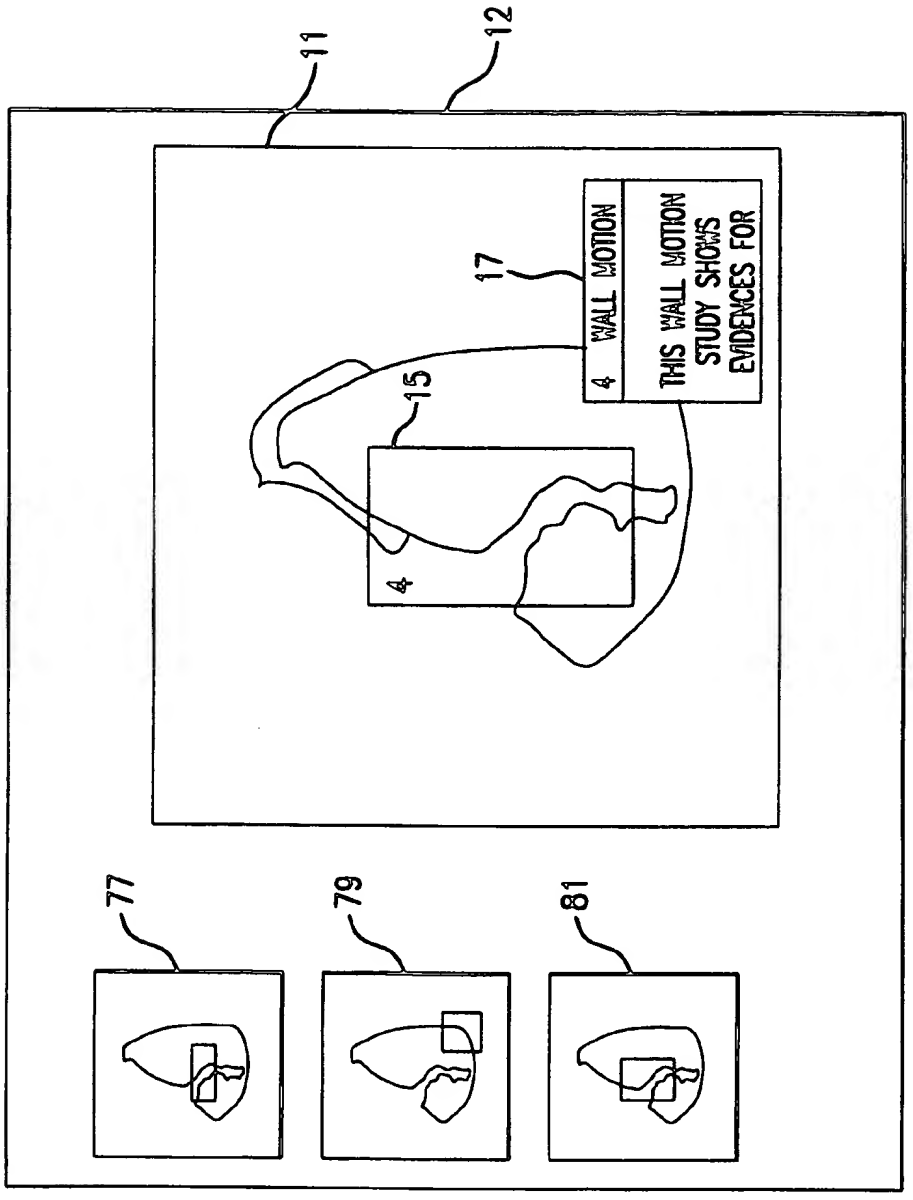
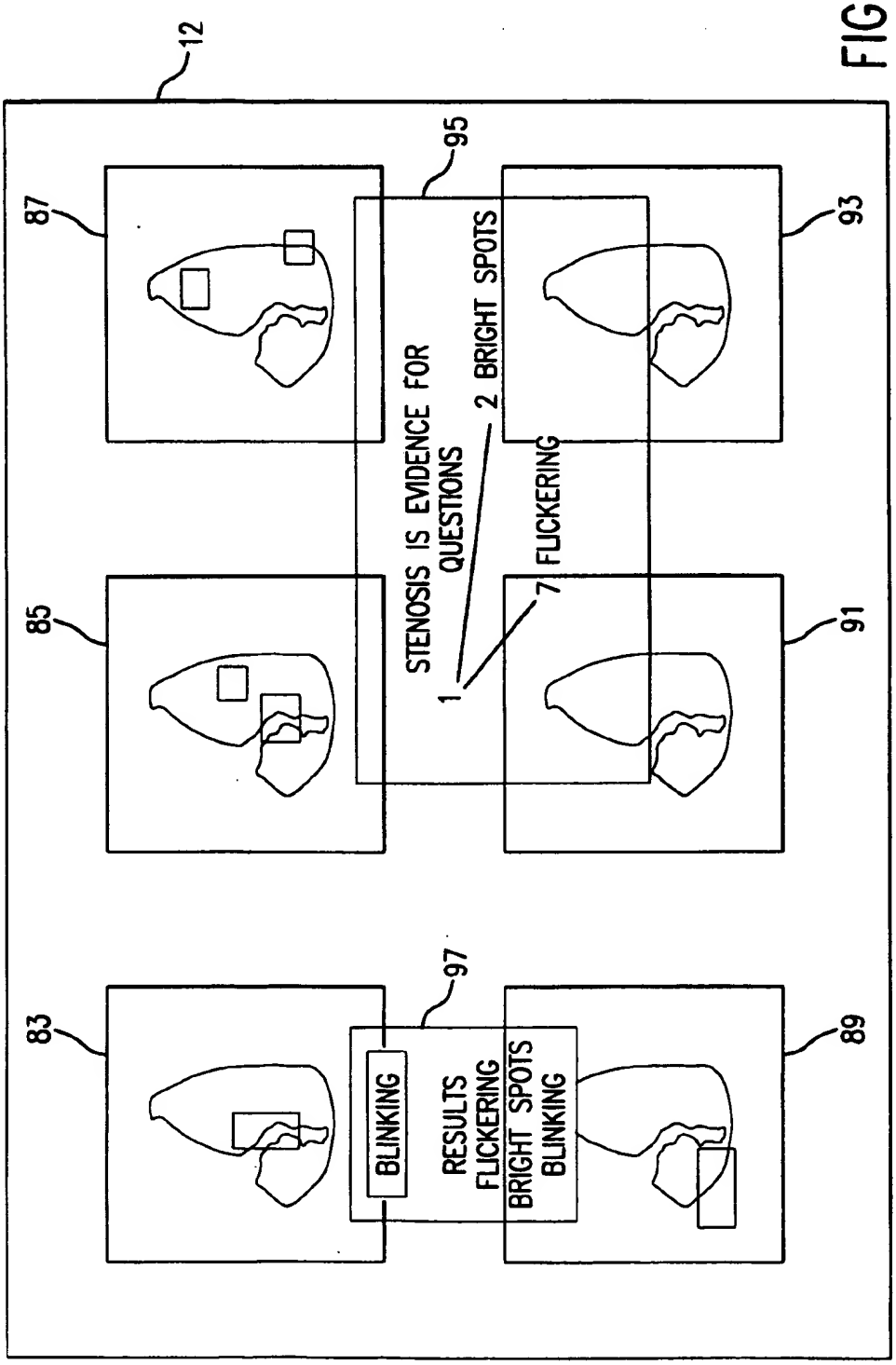

11
12
17
4 WALL MOTION
THIS WALL MOTION STUDY SHOWS EVIDENCES FOR
15
4
77
79
81

FIG.5

FIG.6


12
87
95
93
2 BRIGHT SPOTS
STENOSIS IS EVIDENCE FOR QUESTIONS
FLICKERING
85
1
7
91
83
97
BLINKING
RESULTS FLICKERING BRIGHT SPOTS BLINKING
89

ITERATIVE VIDEO TEACHING AID WITH RECORDABLE COMMENTARY AND INDEXING

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to a teaching or diagnostic aid and/or knowledge recording system whereby images or portions of images may be selected, and labeled, and concurrent image-specific commentary applied to the visual image to create a teaching session which is indexable for later study, creation of further sessions, or research.

2. Description of the Related Art

In the diagnostic or teaching environment, such as a hospital or a medical school, great technological changes have taken place in terms of information storage, imaging, recording, and display technology. Yet the way in which knowledge is passed on has changed little. Data, or case, interpreters and learners meet in small groups and discuss individual case studies with a teacher or each other, to share knowledge and gain experience. When such sessions take place, the experiential knowledge is shared in informal and sometimes haphazard ways and only through a large number of repeated sessions is the individual's knowledge increased. Especially where the case study is in motion, e.g., a video presentation of biological events occurring in sequence over a time line, the record of such a teaching or diagnostic session is hard to establish with sufficient clarity in order to allow future users the benefit of a review of the session.

Particularly in the field of internal medicine, imaging technology has revolutionized our understanding of abnormalities in biological and neurological systems. Fast changing technological improvements, impacting the quality of the images, make it even more difficult to compare cases and build an understanding of what the normal state is. The processes for combining data interpretation, documentation and retrieval are currently inefficient or lead to a high degree of abstraction in the diagnostic record. Despite all efforts in automated image recognition, the human brain remains the best system for pattern recognition. Therefore, it would be desirable to match an image data base with appropriate indexing to the diagnostic abilities of the human data interpreter.

By way of illustration, in the medical field, a cardiologist may review live or recorded echocardiograms with medical students on a video screen in an informal session as need or opportunity arises. Deictic gestures will usually include pointing to particular areas of the screen as the cardiac events happen in time. A commentary about the image will accompany each gesture. However, when the session ends the externalized knowledge expressed in gestures and accompanying commentary during the session is largely unrecorded and unavailable for future use. All that is left is fingerprints on the screen.

Known diagnostic systems utilizing video recording and commentary are largely concerned with the image recording and display of the biological events to further automate diagnosis of the event and location of problem areas. Systems such as detailed in U.S. Pat. No. 5,619,995 to Lobodzinski or in U.S. Pat. No. 5,235,510 to Yamada et al. are sophisticated tools of medical practice but do not appear to disclose critical aspects of a teaching, or interactive interpretation, system suitable for every day use in a real world, informal, teaching hospital environment.

What is needed in such environments, and many other informal teaching or diagnostic environments, is a system whereby all aspects of a teaching, or diagnostic, session occurring around an in-motion event, including gestures and commentaries of the students and teachers, may be recorded in time-synchronized fashion to capture the knowledge expressed during the session for later use. It is desirable that such a system be easily learned and usable without special techniques. It is also desirable that the system be utilizable with a variety of video formats so as to be applicable to past and present recorded image data formats. It is further desirable that the knowledge expressed be cataloged, or indexed, for convenient retrieval and for comparison to contrasting, or similar, case studies to further heighten the efficiency of learning and sharing of experiential knowledge.

DEFINITIONS

"Frame" or "full frame" refers to an image covering the entire display screen.

"Inserted frame" refers to a full image displayed as a smaller portion inserted into the full frame analogous to a "window" or "picture in picture".

"Subframe" refers to a portion of a fill image as selected by the viewer.

"Concurrent commentary" refers to comments made at the time of, or relating to, the subframe image selection.

SUMMARY OF THE INVENTION

The present invention teaches method and apparatus for recording experiential knowledge desirably including playback of in-motion events (video), the recording of deictic gestures linked to particular images in the video, and the linking of commentary accompanying the deictic gestures, with the video images. The commentaries may then be indexed for easy retrieval and for cross-reference to images bearing on a similar topic.

In general the present invention is accomplished by overlaying an interpretation layer on the video, or image layer, to record the deictic gestures of a viewer in time and location synchronization with the video, through a software application. The interpretation layer will also record comments and associate the comments with the gestures. Put in other words, the interpretation layer will create a gesture map and comment map overlaid on the video sequence with all three being synchronized to the video, whether by time code, frame and line code, or other image identifier accompanying the video. The present invention is most efficacious with in-motion images but may work equally well with a series of static images as the image layer.

Deictic gestures may be registered by any suitable means such as mouse operated cursor, a touch screen mechanism, light pen, or other means, now known or later developed, as dictated by the functional and economic considerations of the builder or user. Commentary may be continuously recorded and coded with the video and gestures, or the act of recording a gesture may open a window for comment specific to the gesture. Commentary may be entered vocally, through a keyboard, or through a menu of comment options, or any other means suitable to the application. The video may be paused, or looped between certain frames, to more easily accommodate the user in placing more than one gesture tag in a frame. The gestures and indexed commentary may be labeled a "session" and will be indexed appropriately. Later sessions having additional gestures and commentary may be built upon a previous session.

By replaying the session, a later viewer is privy to the deictic gestures and accompanying comments of previous

viewers thereby gaining a much more thorough understanding of the experiential knowledge, or areas of question, or both, expressed by the prior viewers. By linking through the index, the viewer may access any number of data bases to enhance their appreciation of the subject matter of the present session, further enhancing the value of the educational experience.

BRIEF DESCRIPTION OF THE DRAWINGS

FIGS. **1A–1D** and **2A–2C** illustrate teaching or diagnostic sessions of the present invention.

FIG. **3** is a schematic representation of an apparatus according to one embodiment of the present invention.

FIG. **4** illustrates a video frame from a teaching session with tagged, or selected and labeled, subframes and their associated commentary.

FIG. **5** illustrates a frame from a decision-support session where images of comparable cases are displayed simultaneously with the case under study.

FIG. **6** illustrates a frame from a research session where images from multiple cases are recalled by index topics from a data base.

DESCRIPTION OF PREFERRED EMBODIMENTS

A preferred embodiment will be described in the context of a medical teaching or diagnostic (hereinafter simply referred to as "teaching") aid for the recording of echocardiogram review sessions. The ordinarily skilled artisan will appreciate that the present invention is applicable to teaching or review sessions in many other fields such as criminology, history, or the like, and should not be taken as limited to the discussed context, which is set forth by way of example only.

Referencing FIGS. **1A–1D**, a brief portion of a teaching session is shown to illustrate some principles of the present invention. A full motion video event (hereinafter referred to simply as "video"), such as an echocardiogram, is understood to be occurring on a monitoring system. An image of interest to the image interpreter, e.g. a teacher or diagnostician, occurs on the echocardiogram in FIG. **1A**. The teacher points to the image, and deictically gestures by touching at ref. no. **13** a particular area of the image designated here as a "subframe" **15** (FIG. **1B**). The touch is sensed by a touch screen mechanism as known in the art and a preselected area, or an area defined by the gesture, is highlighted. The motion of the video can be paused or looped if desired. A comment window **17** is then opened in association with the selected, and concurrently labeled or "tagged", subframe, allowing comment text **19** to be entered concerning the selected subframe **15**. If desired, the subframe **15** may be enlarged for further scrutiny, additional emphasis, or the like, as at FIG. **1D**. Motion of the echocardiogram video is then resumed, if it has been stopped, after comment has been entered under a command from the teacher. Each deictic gesture and synonymously labeled commentary are recorded in synchronization with the video for later use.

FIGS. **2A–2C** illustrate playback of the first session illustrated in FIGS. **1A–1D**, and the ability of the system to add additional comments to a previously tagged subframe, or add additional subframes and comments, thereby creating another session. In FIG. **2A** the replay of the recorded first session is started, showing the image **11**. At FIG. **2B** the first subframe **15** and its associated comment has appeared indicating to the viewer the objects of interest and commentary therein. At this point the subframe **15** may again be selected by deictic touch, as at ref. no. **21**, and additional commentary added as at ref. no. **23** (FIG. **2C**). Also, additional subframes, e.g., **25**, may be selected with attendant comment windows **27** and comment entry. All new gestures and commentary are recorded and desirably labeled as a separate session. All commentary will desirably be indexed to aid in educational pursuit, or other decision making processes, as further explained below.

Referencing FIG. **3**, exemplary apparatus for accomplishing sessions according to the present invention will include a computer **29** for control of the video and session maintenance; a display means **31**, e.g., a monitor with a CRT **33** and a speaker **35**, linked to the computer **29**; and deictic gesture producing or capturing apparatus such as touch screen apparatus **37**, a mouse **39**, or keyboard **41**, all linked to the computer **29**. Also, comment entry apparatus such as a microphone **43** or the keyboard **41** are necessary. The mouse **39** may also be used to enter commentary from preselected menus if the semantics of the discipline for which the teaching is intended allows for same.

The computer **29** will include systems for video processing **45**; the interpretation layer application **47** for operation and maintenance of the sessions; data storage **49**, such as ROM, disks, or the like; an indexing application **51** for the indexing of commentary or images selected within a session; and a database and attendant applications **53** for management of broader teaching functions, such as comparing sessions or searching images, as used to supplement individual sessions. The specific construction of hardware and software applications necessary to practice the teachings of the present invention are considered to be within the skill of the person having ordinary skill in the art and therefore such specifics are left to the discretion of the individual artisan. An antenna **55** is shown to schematically indicate that sessions may be remotely sent, accessed, or shared by distance-learning techniques now known or later developed, to facilitate remote collaborative data interpretation sessions. The present invention may also include communication ports for interfacing with various networked data and/or image management systems if desired.

As indicated in FIG. **3**, the video processor **45** will accept input from a variety of sources **57** of live or recorded video sources, including those transmitted from remote sources, and will time code **59** or otherwise quantify the video layer as necessary. Video may be reformatted, digitized, or the like within the scope of the present invention. The video layer time code **59** is shared with the interpretation layer **47** to synchronize the two. The interpretation layer **47**, which governs gesture and comment recordation, is overlayed on the video layer as indicated schematically by separate input lines **61**, **63** for video layer and interpretation, respectively. The overlay methodology is used to simplify the acceptance of multiple types of video source **57** and to protect the original video source from any data corruption. A "session" input line **65** is shown to schematically indicate capture and return of gestures and comments from the display means **31** to the computer **29**.

Referencing FIG. **4**, a frame from a teaching or diagnostic session is shown. The image **11** is displayed in totality as an inserted frame in one section of the video frame **12**. Selected and labeled first, second and third subframe images or indications **67**, **69**, and **71** respectively, are overlaid by the interpretation layer on top of the video frame **12**. A comment box **73** associated and labeled synonymously with the second sub-frame **69** displays the commentary created at the

time the subframe 69 was selected on the image display. A concept map 75 is displayed as a series of text within the video frame 12 to the left of the image 121. The text may be highlighted to indicate current status of the image display topic if desired. The concept map 75 may also be used as a menu to select commentary to be applied to selected subframes which can then be easily indexed according to the text by a database management application for later retrieval. Also, it is possible to build or modify the concept map architecture according to user input, such as the frequency of occurrence of the linking of certain concepts, inquiries, or the like. Automated generation of reports utilizing the diagnostic findings captured by a session according to the present invention is also envisioned.

The concept map may also be used as a tool for calling up additional data associated with the image. Touch screen or cursor selection by a mouse are two exemplary methods of operating the concept map 75. The concept map 75 may also be linked to additional topics by a hierarchical tree, lattice, or semi-lattice of associated terminology for additional association, recordation, and indexing of a session. In alternative embodiments, voice recognition may be used to capture and index natural language commentary with the associated deictic gestures and labeling applied during commentary.

Further referencing FIG. 5, a video frame as used for a decision support mode of the system of the present invention is shown. The video frame 12 includes the image of interest 11 in a large format inserted frame to the right of frame 12 and displays a labeled subframe 15 and its associated comment window 17 as overlaid on the video image layer by the interpretation layer. To the left of the large format inserted frame image 11 are displayed multiple, e.g. first, second, and third, small format inserted frames 77, 79, and 81 of similar images from different sessions to be used for comparison to the image of interest in the present session to arrive at a diagnostic conclusion. The small format images are retrieved from the indexed database by keyword, image pattern recognition, or other known, or later developed means or applications. Image size selection and lay-out, such as two images occupying a split screen, may be varied at the control of the operator if desired. Recordation of additional deictic gestures and associated commentary may be added at any time to any part of the frame 12 by the interpretation layer.

Referencing FIG. 6, a video frame as used in a pattern finding or research session mode of the system of the present invention is shown. Within the video frame 12, first through sixth multiple small format images from sessions of similar cases 83, 85, 87, 89, 91, and 93, respectively, are displayed. The sessions will have been retrieved from the index database at the command of the operator. An overlay by the interpretation layer has placed a concept map text box 95, showing a semi-lattice of associated concepts, and an associated query text box 97, from the indexing protocol at the disposal of the operator to aid in the research function. Again all deictic gestures and associated comments may be entered and recorded by the interpretation layer during this mode of operation. Additional analytic applications such as statistical analysis programs, automated report production, or the like may be linked to the research session mode if desired.

Having thus described a teaching system for the utilization of full motion video which captures the flow of thought and expression from live teaching sessions and indexes the same for increased dissemination and utilization of the knowledge expressed; it will be appreciated that many variations thereon will occur to the artisan upon an understanding of the present invention, which is therefore to be limited only by the appended claims.

I claim:

1. A diagnostic and teaching aid including:

a. means for displaying a sequence of images;

b. means for selecting a portion of one image of the sequence of images via gestures of a diagnostician and means for recording a subframe image selection;

c. means for recording a commentary of the diagnostician concerning a selected image portion;

d. means for linking a recorded commentary concerning the selected image portion to the selected image portion;

e. means for recording a plurality of image portion selections and a sequence in which the plurality of image portion selections were selected so as to create a teaching session; and

f. means for indexing at least one of the recorded commentary and a selected image portion so as to make the teaching session retrievable through an index.

2. The diagnostic and teaching aid of claim 1 wherein the means for displaying sequence of images includes a digital video processor.

3. The diagnostic and teaching aid of claim 1 wherein the sequence of images are full motion images.

4. The diagnostic and teaching aid of claim 1 wherein the means for selecting a portion of one image includes a mouse.

5. The diagnostic and teaching aid of claim 1 wherein the means for selecting portion of one image includes a keyboard.

6. The diagnostic and teaching aid of claim 1 wherein the means for selecting portion of one image includes a touch screen.

7. The diagnostic and teaching aid of claim 1 wherein the means for recording commentary includes a menu of terminology related to the images.

8. The diagnostic and teaching aid of claim 7 further including means for associating related terms within the menu of terminology.

9. The diagnostic and teaching aid of claim 8 further including means for recording a frequency with which terms are related by the diagnostician and adjusting the menu of terminology according to the frequency.

10. The diagnostic and teaching aid of claim 1 further including means for indexing the selected image portion by automated image pattern recognition.

11. The diagnostic and teaching aid according to claim 1 further including: means for playing back the sequence of images while displaying the plurality of image portion selections and recorded commentary concerning the selected image portions, the play back being in the sequence in which the plurality of image portion selections were selected.

12. A real time diagnostic and teaching aid including:

a. means for displaying full motion images;

b. means for selecting subframe images from displayed full motion images in real time via gestures of a diagnostician and means for recording a subframe selection;

c. means for recording and linking concurrent commentary of the diagnostician to selected subframe images in real time;

d. means for recording a sequence of subframe selection and linked commentary to create a teaching session;

e. means for indexing the session by at least one of the commentaries and subframe images;

f. means for play back of the session; and

g. means for adding additional commentary into the session.

13. The real time diagnostic and teaching aid according to claim **12** further including means for selecting additional sub-frames during the session playback to create a further session; and means for indexing the further session.

14. The real time diagnostic and teaching aid according to claim **12** further including a separate application layer for overlaying image selection and concurrent commentary on the image data display without disturbing the image data.

15. The real time diagnostic and teaching aid according to claim **12** further including means for displaying more than one session at a time as inserted frames.

16. The real time diagnostic and teaching aid according to claim **12** wherein the means for indexing the further session includes means for indexing by at least one of the concurrent commentary, the additional commentary, and the subframes images.

17. A diagnostic aid or teaching system comprising a computer having:

a. a video layer for the display and quantization of a series of images;

b. an interpretation layer, time synchronized to, and running concurrently with, the video layer, and having a gesture recorder for identifying and recording selected portions of images in the video layer;

c. the interpretation layer further including a comment recorder which records and displays concurrent comments and associates the concurrent comments with the gestures;

d. synchronization means for synchronizing the video layer and the interpretation layer;

e. the interpretation layer overlaid on the video layer so as to be displayed and run concurrently and in synchronization therewith; and

f. an index manager for indexing of the comments or the select portions, or both, within a database; and

g. whereby a diagnostic or teaching session can be created comprising playback of the series of images concurrently with image portions selected by gestures and the comments associated with the gestures.

18. The diagnostic aid or teaching system of claim **17** further comprising: storage means for recording activity within the interpretation application as synchronized to the video application.

19. A method of producing a teaching session with in-motion video comprising:

a. displaying frames or inserted frames of full motion image data;

b. selecting and recording subframe portions of the image data;

c. appending comments to the selected subframe portions of the image data;

steps "a", "b", and "c" making a session;

d. recording the session; and

e. indexing the session.

20. The method of producing a teaching session with in-motion video of claim **19**, further comprising: synchronizing the selecting and recording step and the appending step to the display of the full motion image data.

21. The method of producing a teaching session with in-motion video of claim **20**, further comprising: step a being performed in a separate application from steps b and c to prevent disturbing the full motion image data.

22. The method of producing a teaching session with in-motion video of claim **20**, further comprising: performing steps b) and c) during display of the full motion image data.

23. The method of producing a teaching session with in-motion video of claim **20**, further comprising: playing back the recorded session.

24. The method of producing a teaching session with in-motion video of claim **20**, further comprising: displaying more than one session at a time as inserted frames.

25. A diagnostic and teaching aid method including the steps of:

a. displaying a sequence of images;

b. selecting a portion of one image of the sequence of images via gestures of a diagnostician;

c. recording a commentary of the diagnostician concerning the image portion selection;

d. linking a recorded commentary concerning an image portion selection to an image portion selection;

e. recording a sequence of a plurality of image portion selections to create a teaching session;

f. indexing at least one of the commentary and selected image portions so as to make the teaching session retrievable through an index; and

g. playing back the sequence of images while displaying the plurality of image portion selections and commentary concerning the image portion selections, the play back including the sequence of image portion selection.

* * * * *

US006336029B1

(12) United States Patent
Ho et al.

(10) Patent No.: US 6,336,029 B1
(45) Date of Patent: *Jan. 1, 2002

(54) **METHOD AND SYSTEM FOR PROVIDING INFORMATION IN RESPONSE TO QUESTIONS**

(76) Inventors: **Chi Fai Ho**, 4816 Cabello Ct., Union City, CA (US) 94587; **Peter P. Tong**, 1807 Limetree La., Mountain View, CA (US) 94040

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 40 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **09/596,844**

(22) Filed: **Jun. 19, 2000**

Related U.S. Application Data

(63) Continuation of application No. 09/347,184, filed on Jul. 2, 1999, which is a continuation of application No. 09/139,174, filed on Aug. 24, 1998, now Pat. No. 5,934,910, which is a continuation of application No. 08/758,896, filed on Dec. 2, 1996, now Pat. No. 5,836,771.

(51) **Int. Cl.**[7] **G09B 7/00**

(52) **U.S. Cl.** **434/362**; 434/118; 434/169; 434/323; 704/9; 704/260; 706/927

(58) **Field of Search** ..... 434/118, 156, 434/169, 185, 219, 307 R, 308, 322, 323, 327, 350, 335, 362, 365; 704/9, 270, 102, 3, 8, 260, 258, 265, 266; 701/200, 207, 209; 705/1, 7, 10, 74; 707/1–4, 6, 9, 100, 104.1, 219, 102, 501.1, 500, 532; 706/927; 340/995; 345/810; 455/3.04; 713/155; 725/1, 14, 25, 35, 46, 34, 116, 12 P; 709/219

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,847,784	A	7/1989	Clancey
4,914,590	A	4/1990	Loatman et al.
5,301,314	A	4/1994	Gifford et al.
5,309,359	A	5/1994	Katz et al.
5,404,295	A	4/1995	Katz et al.

(List continued on next page.)

FOREIGN PATENT DOCUMENTS

EP	0 436 459 A	7/1991
WO	WO 93/21587	10/1993
WO	WO 95/02221	1/1995

OTHER PUBLICATIONS

Success Maker Courseware, published by Computer Curriculum Corporation, Dec. 1994.

Active Mind Series from World Wide Web, URL=http://www.broderbund.com/studio/ams.html, 1996.

Logical Journal of the Zoombinis from World Wide Web, URL=http://www.broderbund.com/studio/atoz/zoombini.html, 1996.

Selecting "Software" by PC Novice Series, vol. 3, Issue 12, pp. 51, 64, and 89–92. 1995.

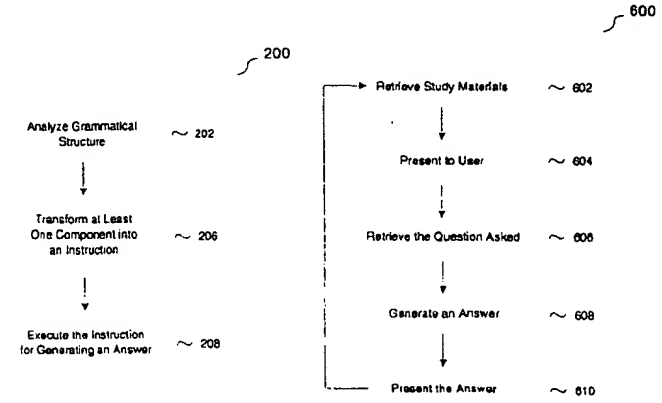
(List continued on next page.)

Primary Examiner—Joe H. Cheng
(74) *Attorney, Agent, or Firm*—Beyer Weaver & Thomas, LLP

(57) **ABSTRACT**

An apparatus and a system to teach a user a subject based on his questions. The system allows the user to control his learning process, and helps to fill in gaps of misunderstanding in the subject. In one embodiment, the system, including a database, presents study materials on the subject to the user. After working on the presented materials, the user enters his question into the system, which generates an answer to the question, and presents it to him. Then the system compares the question with one or more questions previously entered by the user to determine his understanding level in the subject. Based on the determination, the system may present to the user appropriate study materials. The user typically asks more than one question, and the process of answering his question by the system repeats.

86 Claims, 16 Drawing Sheets



US006336029B1

(12) United States Patent
Ho et al.

(10) Patent No.: US 6,336,029 B1
(45) Date of Patent: *Jan. 1, 2002

(54) **METHOD AND SYSTEM FOR PROVIDING INFORMATION IN RESPONSE TO QUESTIONS**

(76) Inventors: **Chi Fai Ho**, 4816 Cabello Ct., Union City, CA (US) 94587; **Peter P. Tong**, 1807 Limetree La., Mountain View, CA (US) 94040

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 40 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **09/596,844**

(22) Filed: **Jun. 19, 2000**

Related U.S. Application Data

(63) Continuation of application No. 09/347,184, filed on Jul. 2, 1999, which is a continuation of application No. 09/139,174, filed on Aug. 24, 1998, now Pat. No. 5,934,910, which is a continuation of application No. 08/758,896, filed on Dec. 2, 1996, now Pat. No. 5,836,771.

(51) **Int. Cl.**[7] **G09B 7/00**

(52) **U.S. Cl.** **434/362**; 434/118; 434/169; 434/323; 704/9; 704/260; 706/927

(58) **Field of Search** ..... 434/118, 156, 434/169, 185, 219, 307 R, 308, 322, 323, 327, 350, 335, 362, 365; 704/9, 270, 102, 3, 8, 260, 258, 265, 266; 701/200, 207, 209; 705/1, 7, 10, 74; 707/1–4, 6, 9, 100, 104.1, 219, 102, 501.1, 500, 532; 706/927; 340/995; 345/810; 455/3.04; 713/155; 725/1, 14, 25, 35, 46, 34, 116, 12 P; 709/219

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,847,784	A	7/1989	Clancey
4,914,590	A	4/1990	Loatman et al.
5,301,314	A	4/1994	Gifford et al.
5,309,359	A	5/1994	Katz et al.
5,404,295	A	4/1995	Katz et al.

(List continued on next page.)

FOREIGN PATENT DOCUMENTS

EP	0 436 459 A	7/1991
WO	WO 93/21587	10/1993
WO	WO 95/02221	1/1995

OTHER PUBLICATIONS

Success Maker Courseware, published by Computer Curriculum Corporation, Dec. 1994.

Active Mind Series from World Wide Web, URL=http://www.broderbund.com/studio/ams.html, 1996.

Logical Journal of the Zoombinis from World Wide Web, URL=http://www.broderbund.com/studio/atoz/zoombini.html, 1996.

Selecting "Software" by PC Novice Series, vol. 3, Issue 12, pp. 51, 64, and 89–92. 1995.

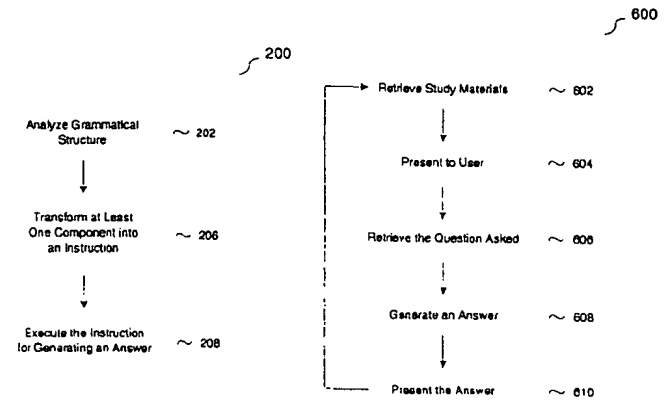
(List continued on next page.)

Primary Examiner—Joe H. Cheng
(74) *Attorney, Agent, or Firm*—Beyer Weaver & Thomas, LLP

(57) **ABSTRACT**

An apparatus and a system to teach a user a subject based on his questions. The system allows the user to control his learning process, and helps to fill in gaps of misunderstanding in the subject. In one embodiment, the system, including a database, presents study materials on the subject to the user. After working on the presented materials, the user enters his question into the system, which generates an answer to the question, and presents it to him. Then the system compares the question with one or more questions previously entered by the user to determine his understanding level in the subject. Based on the determination, the system may present to the user appropriate study materials. The user typically asks more than one question, and the process of answering his question by the system repeats.

86 Claims, 16 Drawing Sheets



U.S. PATENT DOCUMENTS

5,446,883	A		8/1995	Kirkbride et al.
5,454,106	A		9/1995	Burns et al.
5,519,608	A		5/1996	Kupiec
5,581,664	A		12/1996	Allen et al.
5,586,218	A		12/1996	Allen
5,652,828	A		7/1997	Silverman
5,677,835	A		10/1997	Carbonell et al.
5,701,399	A		12/1997	Lee et al.
5,754,938	A	*	5/1998	Herz et al. 725/116
5,754,939	A	*	5/1998	Herz et al. 455/3.04
5,787,234	A		7/1998	Molloy
5,794,050	A		8/1998	Dahlgren et al.
5,797,135	A		8/1998	Whalen et al.
5,835,087	A	*	11/1998	Herz et al. 345/810
5,852,814	A		12/1998	Allen
5,884,302	A		3/1999	Ho
5,909,679	A	*	6/1999	Hall 707/4
5,933,531	A		8/1999	Lorie
5,933,816	A		8/1999	Zeanah et al.
5,959,543	A		9/1999	LaPorta et al.
5,963,948	A	*	10/1999	Shilcrat 707/100
5,999,908	A	*	12/1999	Abelow 705/1
6,076,088	A		6/2000	Paik et al.
6,088,717	A		7/2000	Reed et al.

OTHER PUBLICATIONS

Computerized Adaptive Testing, Oct. 24, 1994, from World Wide Web, URL=Gopher://Gopher.ETS.org, Oct. 1994.

Innovation and Technologies, Oct. 21, 1994, from World Wide Web, URL=Gopher://Gopher.ETS.org, Oct. 1994.

Interactive Mathematics Instructor's Guide by Academic Systems, pp. 86 and 114, Aug. 1995.

"High School Learning and Resource Library" by ProOne, photocopy of the box and screen–dump to get 5 pages, 1995.

Woods, W.A., "Semantics And Quantification In Natural Language Question Answering," AcadEmic Press, pp. 205–248, 1978.

Whalen, Thomas, "*Computational Behaviourism Applied to Natural Langauge*," Communications Research Centre, Ottawa, ON, Apr. 1996, 29 pages.

Bunt, Harry C., "*Dialog Control Functions and Interaction Design*, " Institute for Language Technology and Artificial Intelligence ITK, The Netherlands (in Dialog and Instruction: Modeling and Interaction in Intelligent Tutoring Systems, Heidelberg: Springer Verlag), pp. 197–214, 1995.

Whalen, Thomas, "*CHAT Conversational Hypertext Access Technology*," webpage: http://debra.dgbt.doc.ca/chat/info.page.html, Apr. 1993.

"*The CHAT Natural Language System*," webpage: http://debra.dgbt.doc.ca/chat/chat.html, Dec. 1999.

Prince, Violaine M., "*Relying on a Sophisticated Student Model to Derive Dialog Strategies in an Intelligent Tutoring System*," Dialog and Instruction: Modeling and Interaction in Intelligent Tutoring Systems, Heidelberg: Springer Verlag, pp. 179–194, 1995.

Kaplan, S. Jerrold, "*Cooperative Responses From a Portable Natural Language Database Query System*," Computational Models of Disclosure, MIT Press, Chapter 3, pp. 167–201, 1983.

Norvig, Peter, Paradigms Of Artificial Intelligence Programming: Case Studies In Common LISP, "*STUDENT: Solving Algebra Word Problems*," Chapter 7, Morgan Kaufmann Publishers, pp. 219–235, 1992.

Hendrix et al., "*Developing a Natural Language Interface to Complex Data*," Association for Computing Machinery (ACM), 1978, p. 563–584.

Burton et al., "*Toward a Natural–Language Capability for Computer–Assisted Instruction*," Natural Language Processing, Kaufmann Publishers, pp. 605–624, 1986.

Androutsopoulos et al., "*Natural Language Interfaces to DatabaseS–An Introduction*," Department of Artificial Intelligence, University of Edinburgh, U.K., Mar. 1995, p. 1–50.

Norvig, Peter, "Paradigms Of Artificial Intelligence Programming: Case Studies In Common LISP, "*STUDENT: Solving Algebra Word Problems*," Chapter 7," Morgan Kaufmann Publishers, 1992, pp. 219–235.

Response Generation, "*Question–Answering Systems*," Chapter 16, and "*Natural Language Generation*," Chapter 17, pp. 468–513, before 1996.

Burke et al., "*Knowledge–based Information Retrieval from Semi–Structured Text*," The Artificial Intelligence Laboratory, The University of Chicago, Nov. 1995.

* cited by examiner

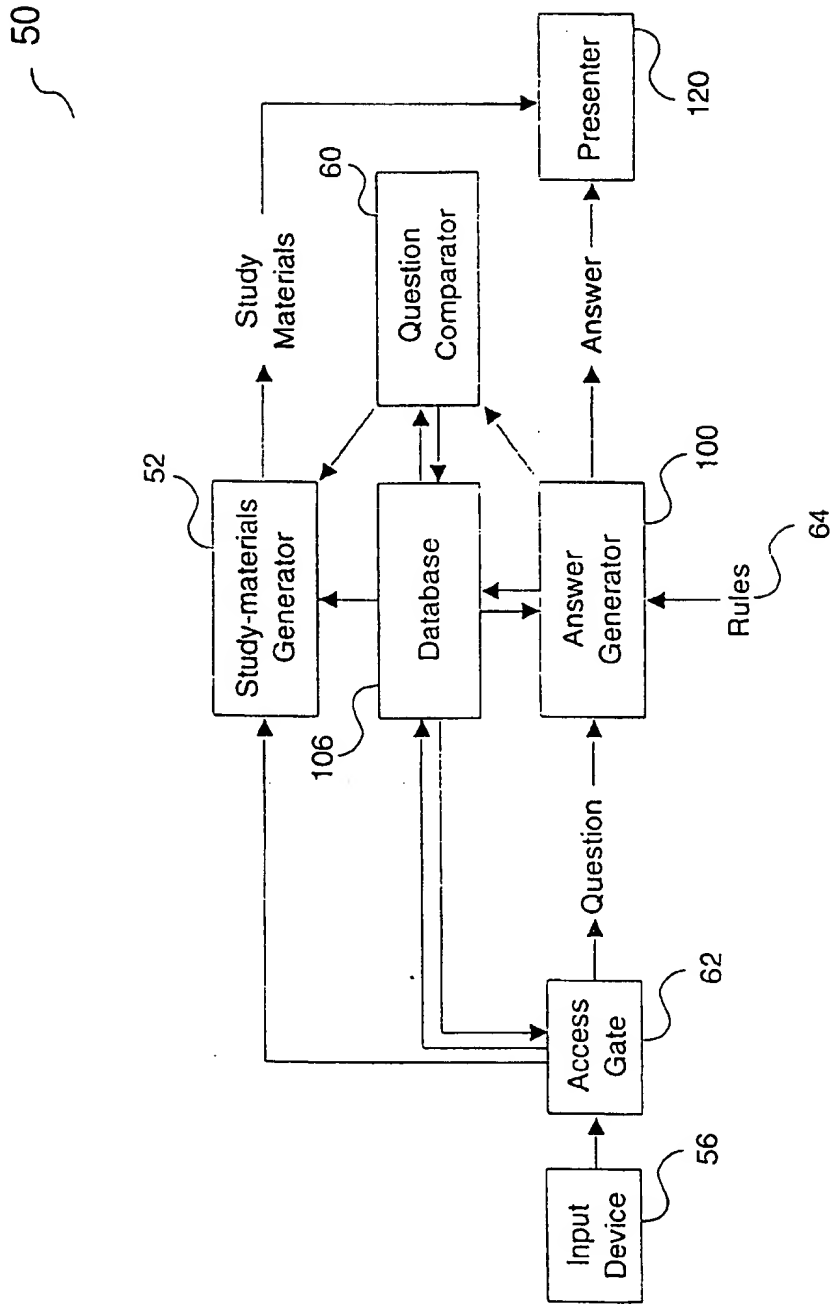

50
Presenter
120
60
Question Comparator
Study Materials
Answer
52
100
64
Study-materials Generator
Database
Answer Generator
Rules
106
Question
62
Access Gate
56
Input Device

Figure 1

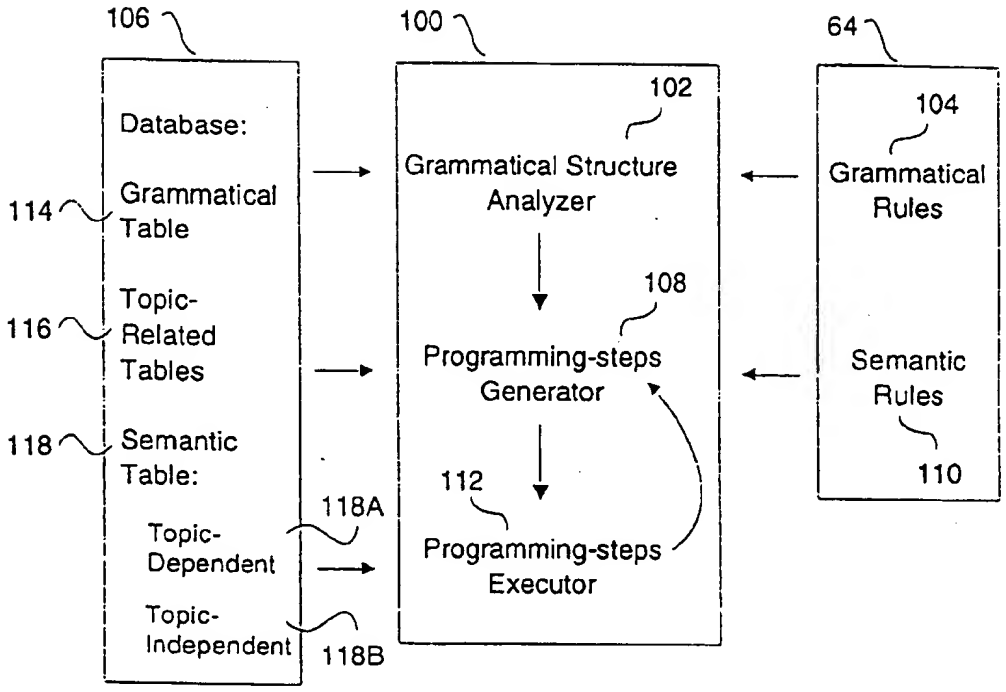

106
100
64
102
104
Database:
Grammatical Structure Analyzer
Grammatical Rules
114
Grammatical Table
108
116
Topic-Related Tables
Programming-steps Generator
Semantic Rules
118
Semantic Table:
110
112
118A
Topic-Dependent
Programming-steps Executor
Topic-Independent
118B

Figure 2

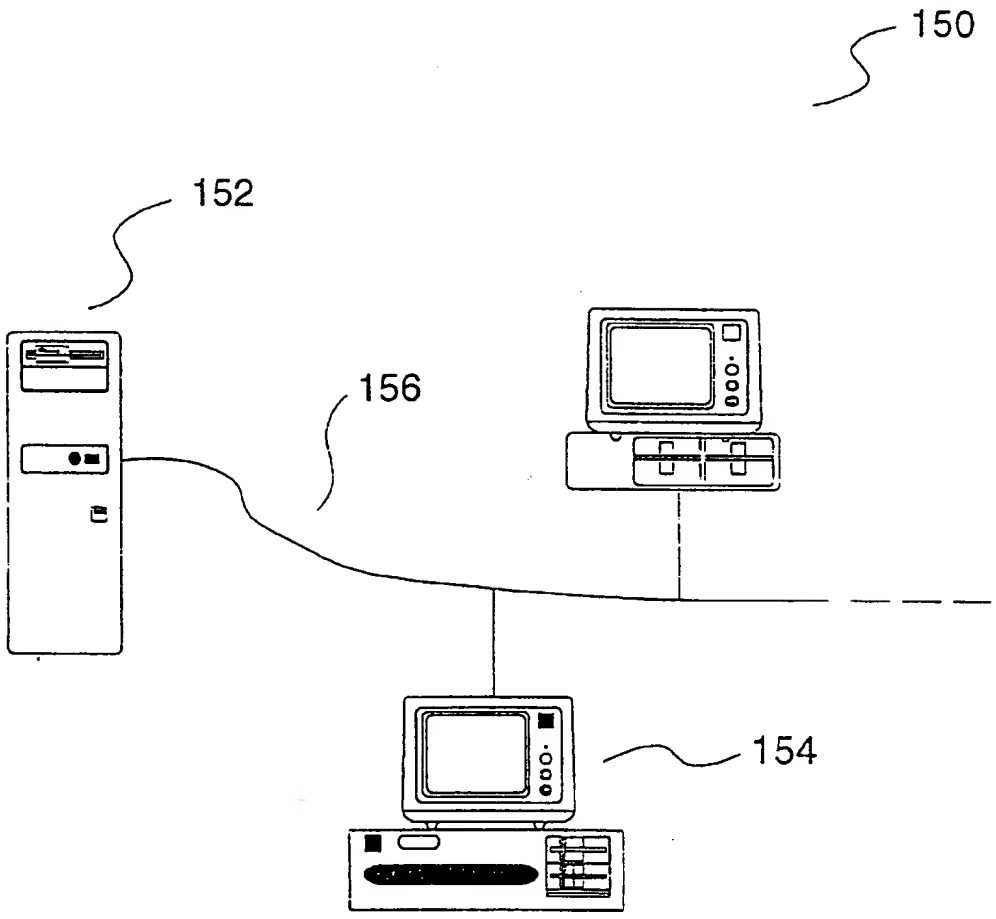


Figure 3A

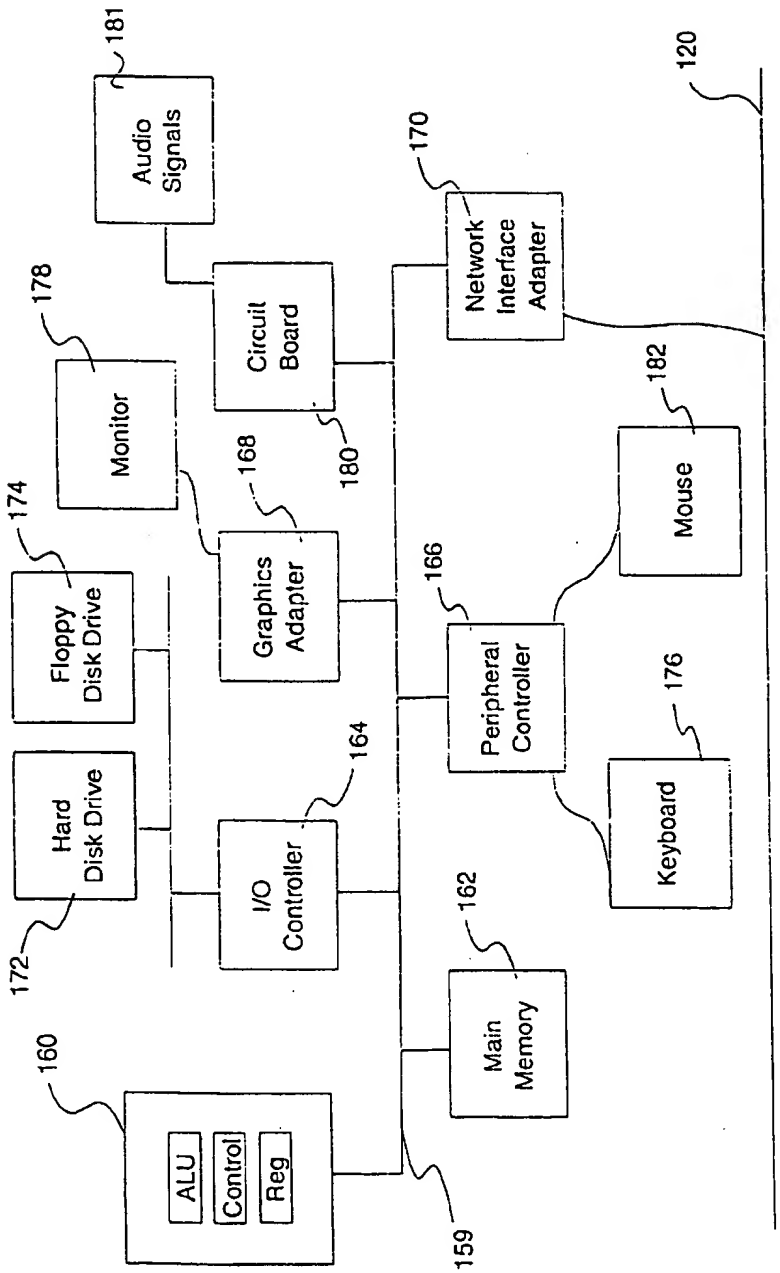

181
Audio Signals
120
170
Network Interface Adapter
178
Monitor
Circuit Board
182
Mouse
168
180
174
Floppy Disk Drive
Graphics Adapter
166
Peripheral Controller
Hard Disk Drive
164
I/O Controller
176
Keyboard
162
172
Main Memory
160
ALU
Control
Reg
159

Figure 3B

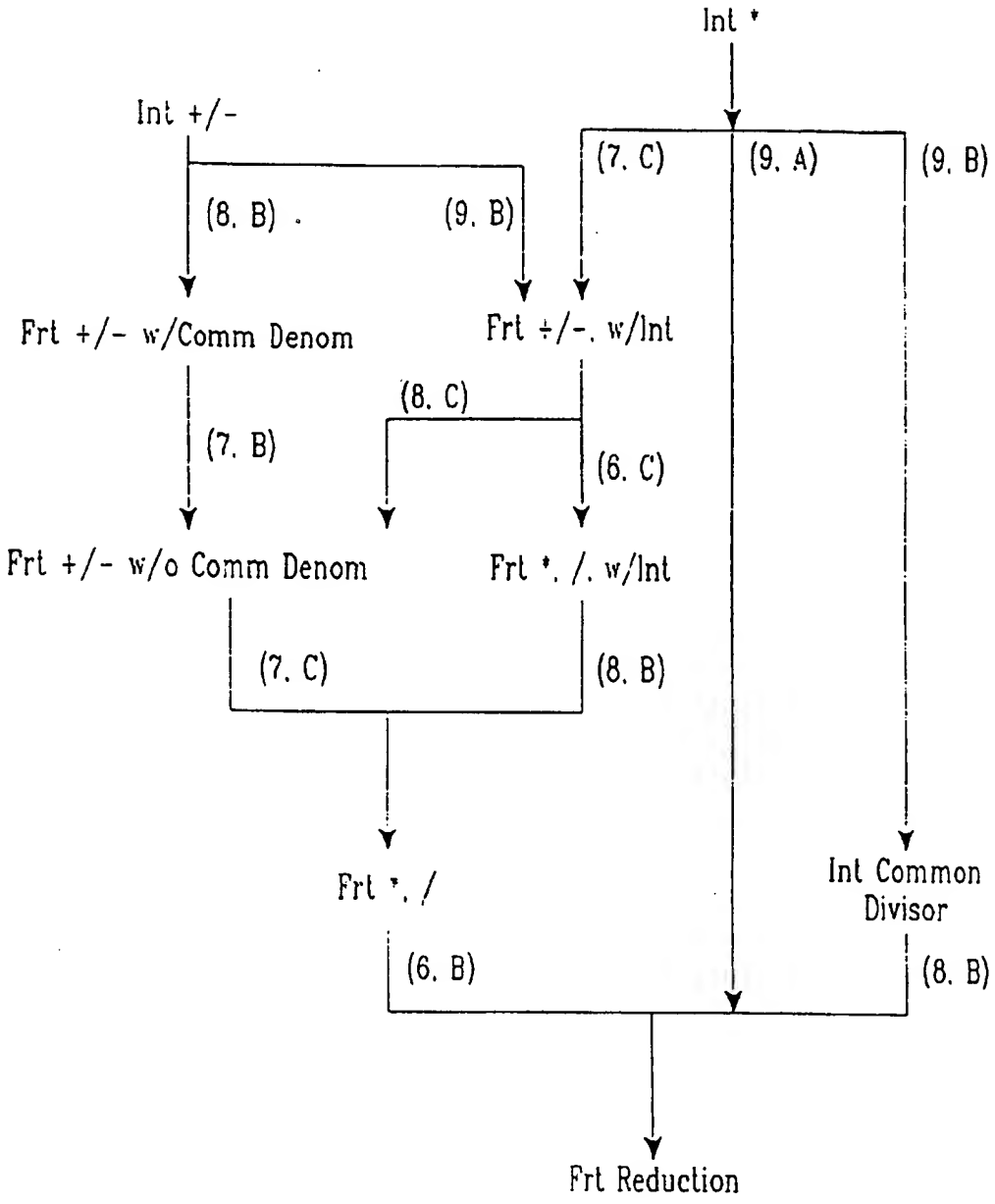

Int *
Int +/-
(7. C)
(9. A)
(9. B)
(8. B)
(9. B)
Frt +/- w/Comm Denom
Frt +/-. w/Int
(8. C)
(7. B)
(6. C)
Frt +/- w/o Comm Denom
Frt *. /. w/Int
(7. C)
(8. B)
Frt *. /
Int Common Divisor
(6. B)
(8. B)
Frt Reduction

Figure 4

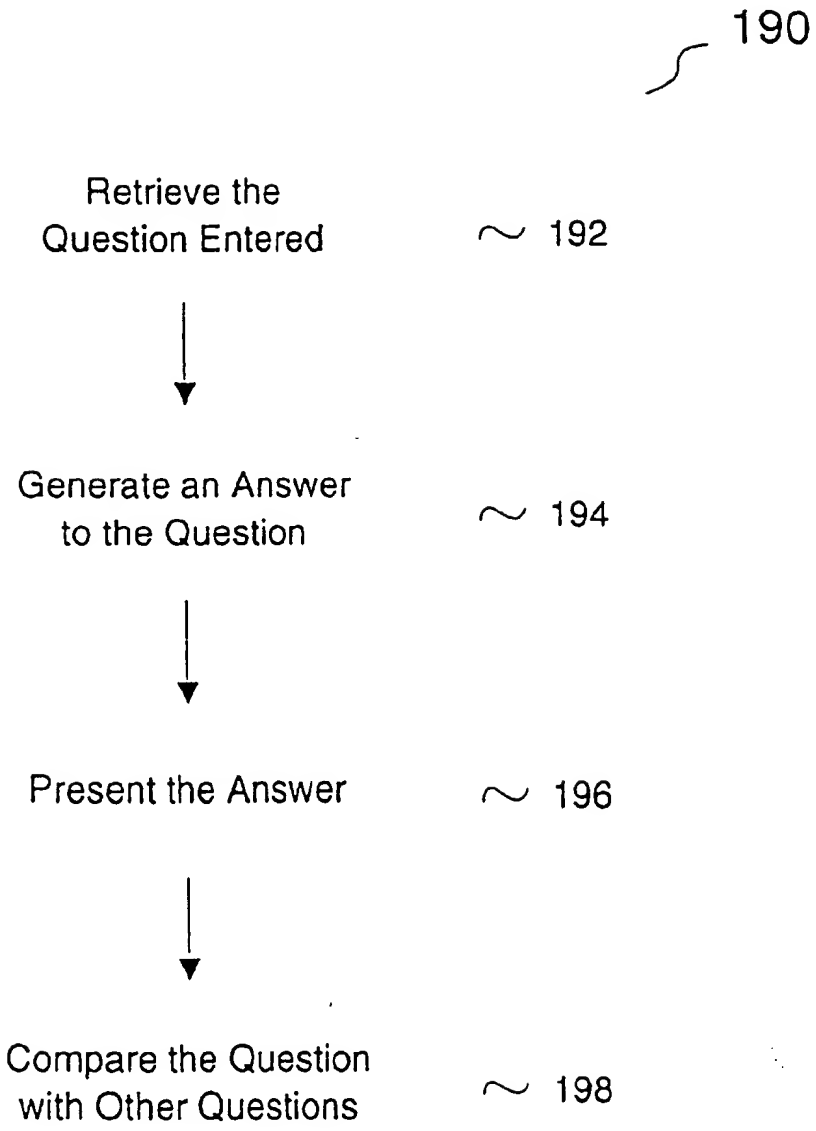

190
Retrieve the Question Entered
192
Generate an Answer to the Question
194
Present the Answer
196
Compare the Question with Other Questions
198

Figure 5

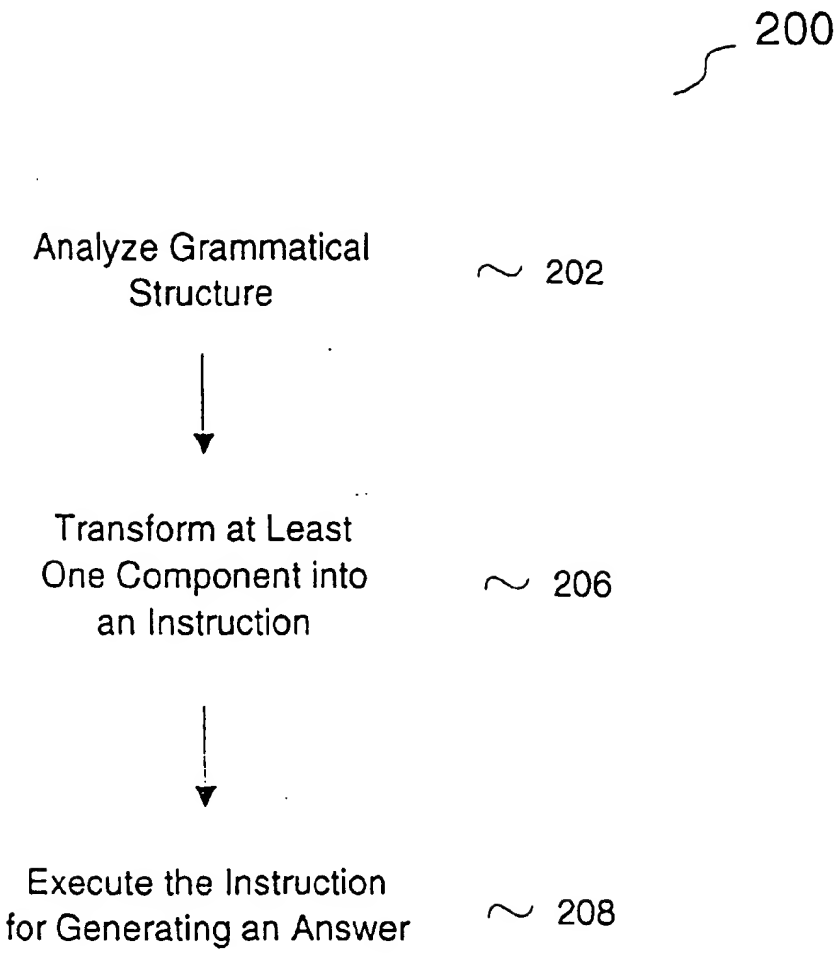

200
Analyze Grammatical Structure
202
Transform at Least One Component into an Instruction
206
Execute the Instruction for Generating an Answer
208

Figure 6

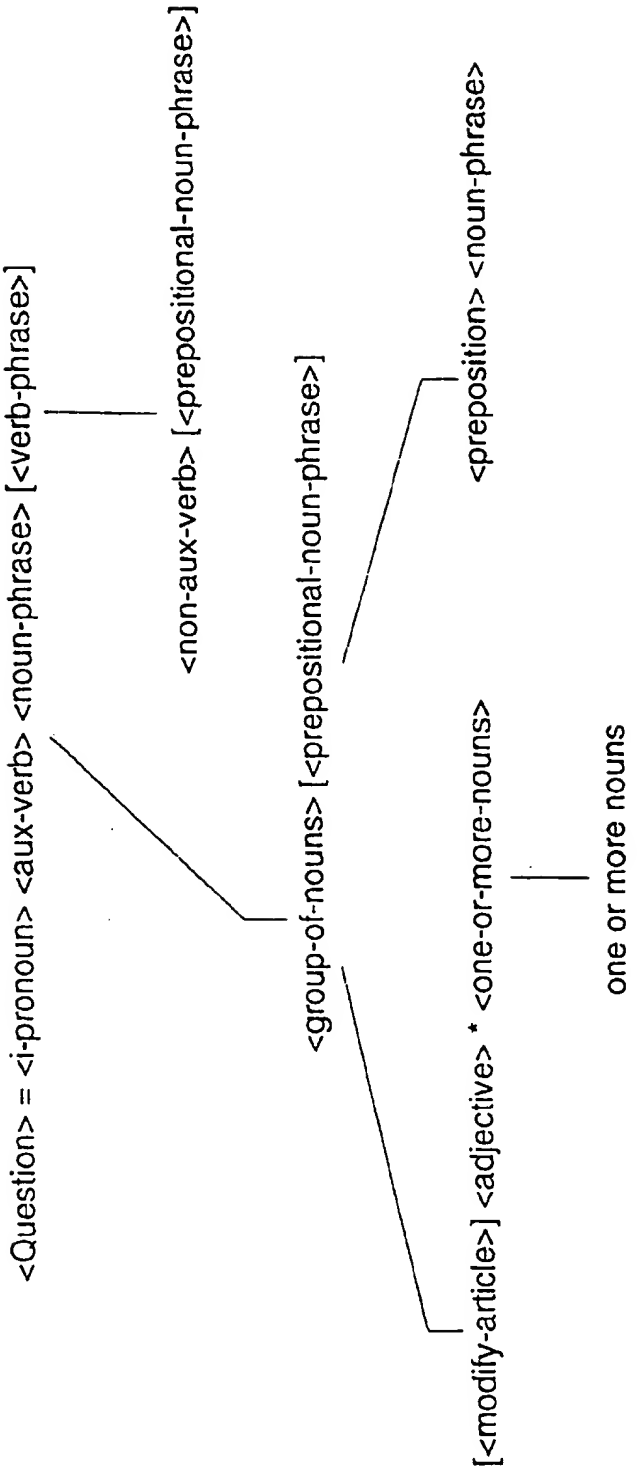

<Question> = <i-pronoun> <aux-verb> <noun-phrase> [<verb-phrase>]
<non-aux-verb> [<prepositional-noun-phrase>]
<group-of-nouns> [<prepositional-noun-phrase>]
<preposition> <noun-phrase>
[<modify-article>] <adjective> * <one-or-more-nouns>
one or more nouns

Figure 7

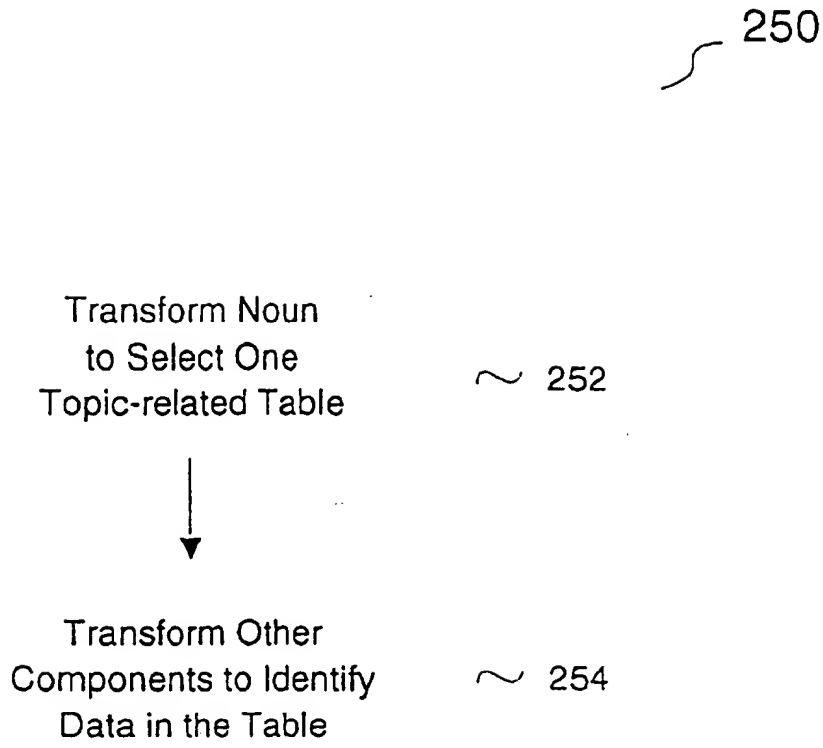

250
Transform Noun to Select One Topic-related Table
252
Transform Other Components to Identify Data in the Table
254

Figure 8A

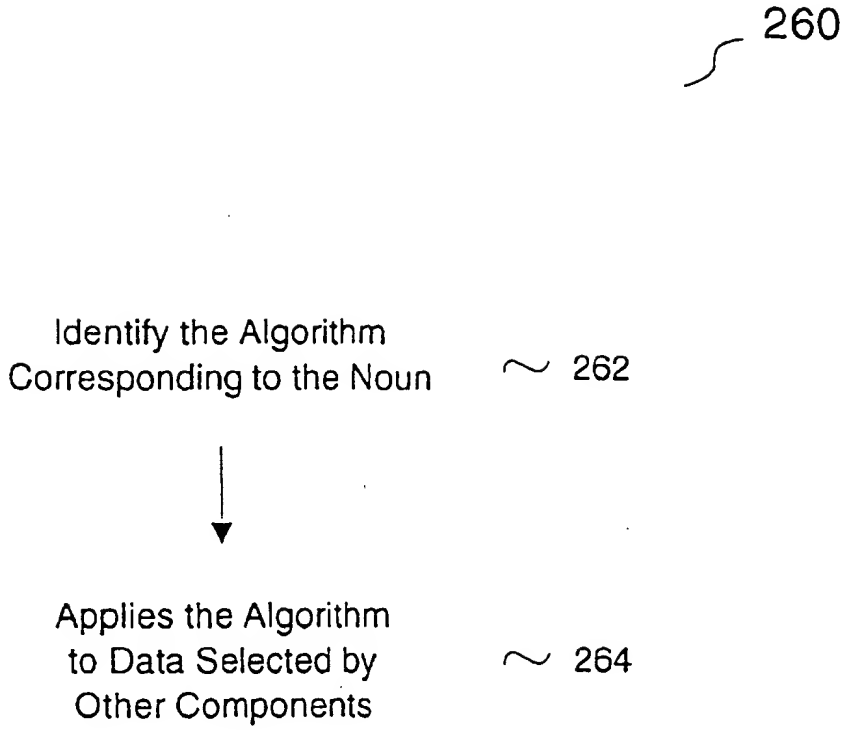

260
Identify the Algorithm Corresponding to the Noun
262
Applies the Algorithm to Data Selected by Other Components
264

Figure 8B

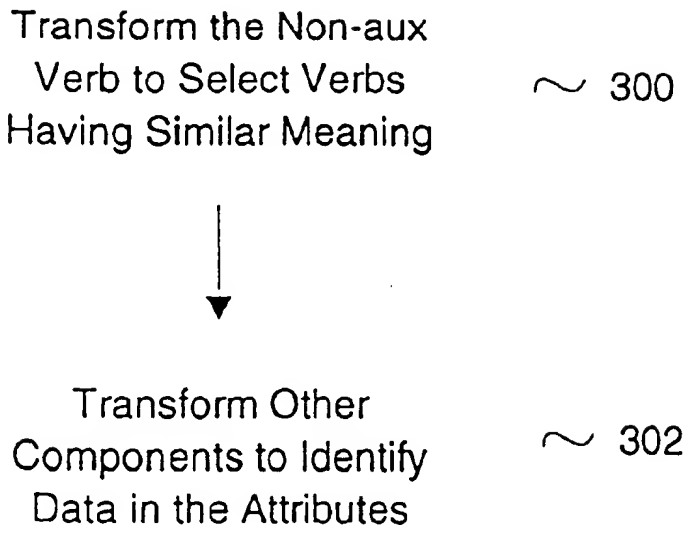

Transform the Non-aux Verb to Select Verbs Having Similar Meaning
300
Transform Other Components to Identify Data in the Attributes
302

Figure 9

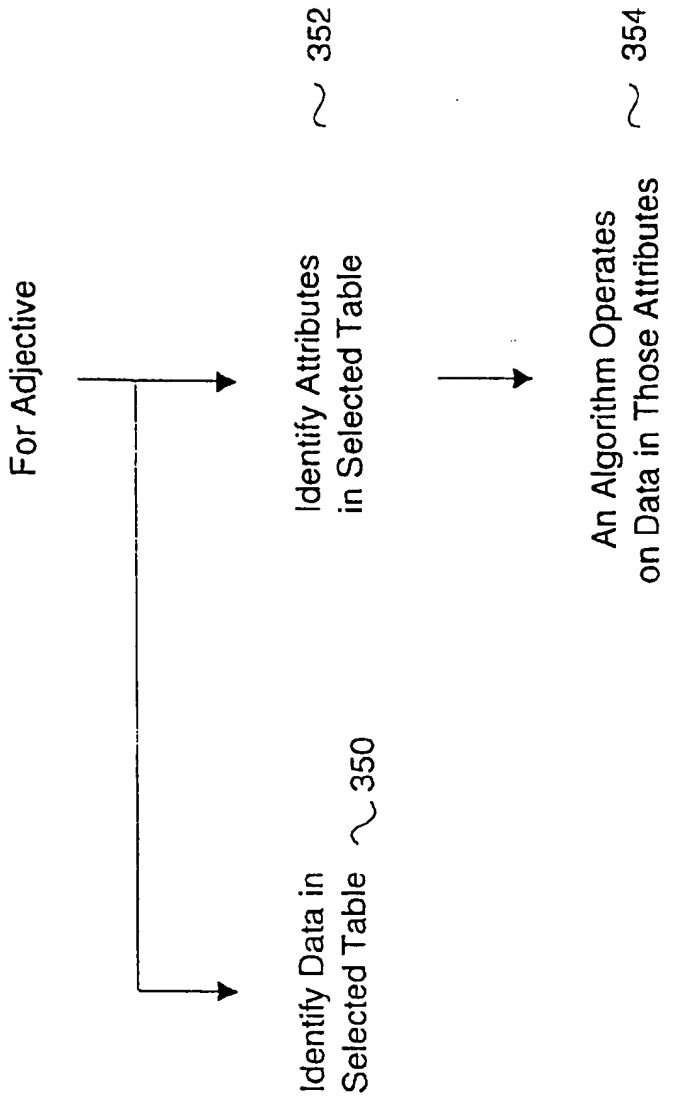

For Adjective
Identify Attributes in Selected Table
352
An Algorithm Operates on Data in Those Attributes
354
Identify Data in Selected Table
350

Figure 10

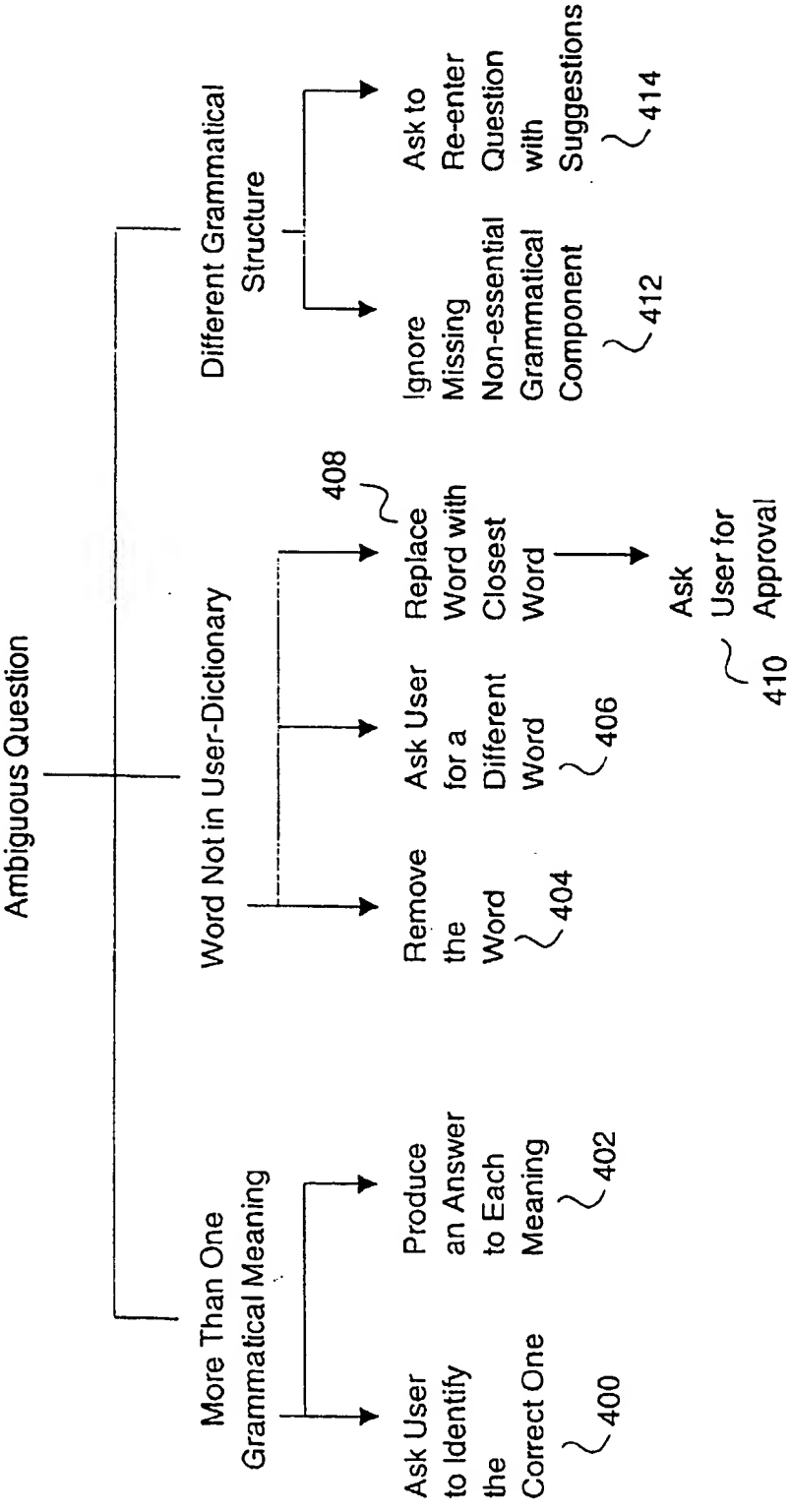

Ambiguous Question
More Than One Grammatical Meaning
Ask User to Identify the Correct One
400
Produce an Answer to Each Meaning
402
Word Not in User-Dictionary
Remove the Word
404
Ask User for a Different Word
406
Replace Word with Closest Word
408
Ask User for Approval
410
Different Grammatical Structure
Ignore Missing Non-essential Grammatical Component
412
Ask to Re-enter Question with Suggestions
414

Figure 11

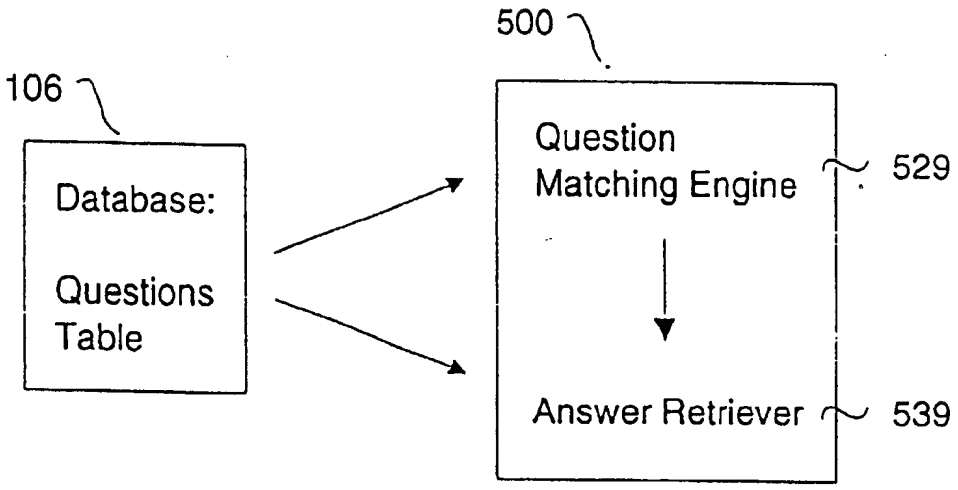

500
106
Database:
Questions
Table
Question
Matching Engine
529
Answer Retriever
539

Figure 12

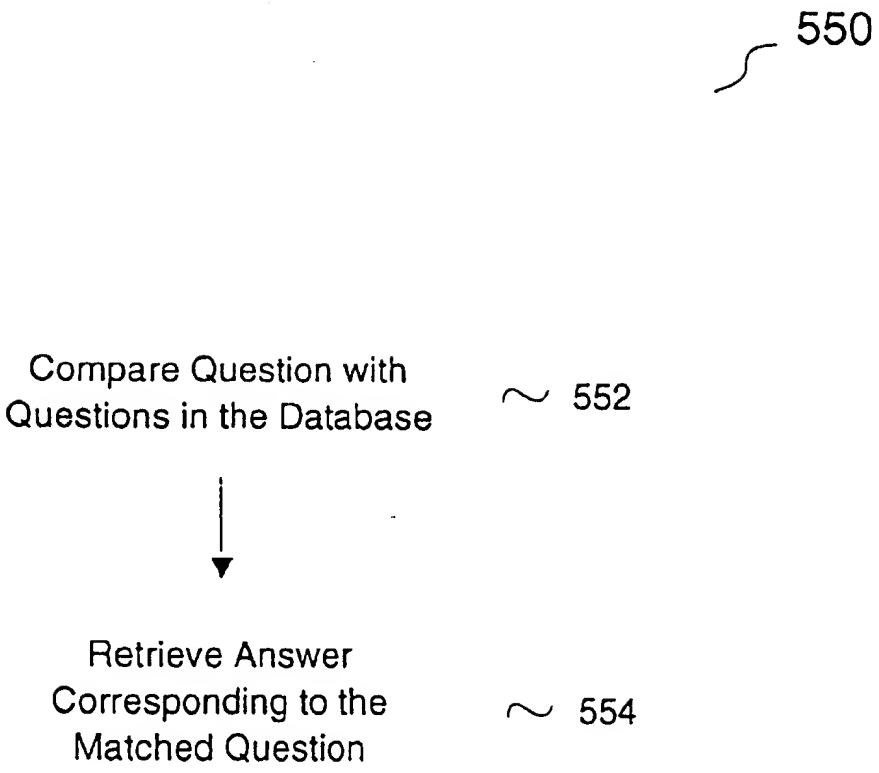

550
Compare Question with Questions in the Database
552
Retrieve Answer Corresponding to the Matched Question
554

Figure 13

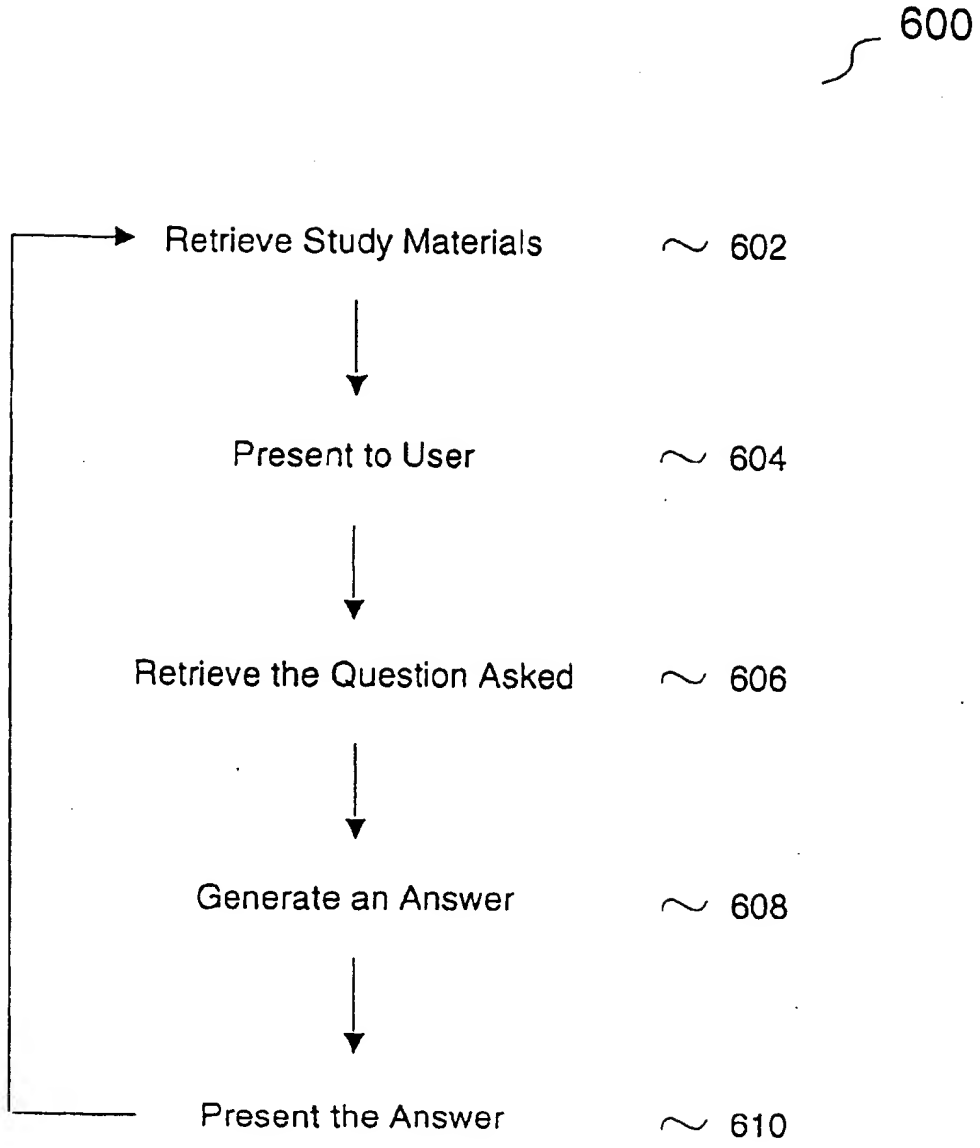

600
Retrieve Study Materials
602
Present to User
604
Retrieve the Question Asked
606
Generate an Answer
608
Present the Answer
610

Figure 14

METHOD AND SYSTEM FOR PROVIDING INFORMATION IN RESPONSE TO QUESTIONS

This application is a continuation of U.S. patent application Ser. No. 09/347,184, filed on Jul. 2, 1999, the disclosure of which is hereby incorporated by reference, which is a continuation of U.S. patent application Ser. No. 09/139,174, filed on Aug. 24, 1998, now U.S. Pat. No. 5,934,910, which is a continuation of U.S. patent application Ser. No. 08/758,896, filed on Dec. 2, 1996, now U.S. Pat. No. 5,836,771; with the applications and patent being incorporated by reference into this application.

BACKGROUND

This invention relates generally to education and, more particularly, to computer learning based on question asking.

Question asking is an important aspect in learning because we have a better understanding in a subject if we can ask questions. As opposed to passive leaning where we just absorb like a sponge, active learning based on asking questions enhances understanding and helps us remember. However, if a person is learning from a computer system, he does not have the luxury of having a question-and-answer dialog with the computer.

Asking questions not only focuses our attention on the subject, it also fills gaps in our understanding. When we are learning from an instructor, typically we cannot comprehend everything. As our misunderstanding grows, very soon we begin to lose track of the subject, and our interest in the subject wanes. Similarly, we lose interest in reading a book with many individuals if we confuse their names. During those instances, asking questions to fill our gaps of misunderstanding might rekindle our interest in the subject or the book.

A user's questions on a subject also indicate how much he understands the subject, If the user repeatedly asks questions in a certain area, he is weak in that area.

In view of the importance of question asking, many instructors include them in teaching. One of the most famous teachers—Socrates—even used questions as his main tool to stimulate thinking and to teach. However, when a computer teaches, the users cannot question the computer the same way he can question his instructor.

Learning through a computer has its benefits. Computer allows a user to learn at his own pace. For a class of thirty, typically the instructor will not hold up the class just to clarify issues for one student. If students' levels of understanding are not the same, the instructor has to leave some of them behind. This problematic situation is prevalent in a classroom with students having different cultural backgrounds and non-uniform understanding levels. Computers can ameliorate such problems. If each student is taught by his computer, he can control the computer so as to learn at his own pace.

However, learning from a computer has its handicap. When the student needs an answer for a question, problem arises because the computer cannot understand his question.

There are computers responding to questions. One is the system to locate books used in many libraries. Users can enter search-requests for books into the system. But such systems are primitive as compared to those where a user can learn a subject by asking questions.

Another system responding to questions is called Elisa. It responds to questions, and tries to emulate a psychiatrist. A user enters a question into Elisa, which changes the entered question around to respond to the user. For example, the user enters, "I feel bad." Elisa might respond, "Why do you feel bad?" The system gets the user to talk, and presumably, the user feels better afterwards. The goal of the system is not to understand the user, but to encourage the user to communicate his problem.

There are also systems that respond to questions written in computer languages. In such systems, the user re-formulates his question into a program to access and to process information from a database. Someone not familiar with programming languages cannot get an answer from those systems.

It should have been obvious that there is a need for a method and a system that can teach a subject through responding to a user's questions.

SUMMARY

This invention is on a method and a system that can teach a subject based on a user's questions. It is different from the user learning a subject through passively absorbing the materials. In this invention, he sets the learning pace, controls the learning process, and can learn by asking questions.

In one embodiment, the system generates study materials that introduce the subject to the user. After studying the presented materials, he can begin asking questions. The system generates an answer to each question, and presents it to him. The system also compares the question with one or more questions previously entered by him. The comparison determines his understanding level in the subject. If the comparison indicates that he is weak in a certain area, the system can present detailed study materials covering those areas. The system also stores the question he just asked, so as to compare to questions he might ask in the future.

Typically the user does not ask one question and stop. He may ask a series of questions to understand the subject. After the system has responded to his questions, based on his understanding level, the system may present to him additional study materials. The process may repeat with him asking additional questions until he understands the subject.

In another embodiment, the user can use the system to fill gaps of misunderstanding in a subject. As he works on the subject through the computer, he encounters areas that he does not understand or he has forgotten. This embodiment allows him to get answers on questions in those areas.

Other aspects and advantages of this invention will become apparent from the following detailed description, which, when taken in conjunction with the accompanying drawings, illustrates by way of example the principles of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 shows one embodiment of the present invention.

FIG. 2 shows one embodiment of the answer generator in the present invention.

FIGS. 3A–B show physical elements implementing one embodiment of the present invention.

FIG. 4 illustrates a part of an hierarchy for the line-items under fractions in the present invention.

FIG. 5 shows one embodiment implementing the present invention.

FIG. 6 shows a set of steps to be used by the embodiment shown in FIG. 2.

FIG. 7 shows a pre-defined context-free grammatical structure in the present invention.

FIGS. 8A–B show examples of semantic rules applied to nouns in the present invention.

FIG. 9 shows an example of semantic rules applied to a non-auxiliary verb in the present invention.

FIG. 10 shows examples of semantic rules applied to adjectives in the present invention.

FIG. 11 shows different approaches to resolve ambiguous question in the present invention.

FIG. 12 shows another embodiment of the answer generator in the present invention.

FIG. 13 shows a set of steps to be used by the embodiment shown in FIG. 12.

FIG. 14 shows a set of steps to fill gaps of misunderstanding in the present invention.

Same numerals in FIGS. 1–14 are assigned to similar elements in all the figures. Embodiments of the invention are discussed below with reference to FIGS. 1–14. However, those skilled in the art will readily appreciate that the detailed description given herein with respect to these figures is for explanatory purposes as the invention extends beyond these limited embodiments.

DETAILED DESCRIPTION

FIG. 1 shows one embodiment of a system 50 in this invention. When a user starts working on a subject, a study-materials generator 52 generates introductory study materials using information from a database 106. A presenter 120 presents those materials. After studying the presented materials, the user might need to ask a question. He enters the question into the system 50 through an input device 56. An answer generator 100 retrieves the question and generates an answer based on information from the database 106 and a set of rules 64. The presenter 120 then presents the answer to him. A question comparator 60 also can compare the question with one or more questions he previously asked. Those questions were stored in the database 106. The comparison determines his understanding level in the subject. If the comparison indicates he is weak in a certain area, the study-materials generator 52 will retrieve study materials from the database 106 covering that area. The database 106 also stores the question just asked for future comparison.

FIG. 2 shows one embodiment of the answer generator 100 for answering a natural-language question, which is a question used in our everyday language. In the present invention, a question is defined as an inquiry demanding an answer; and an answer is defined as a statement satisfying the inquiry.

An input device, such as a keyboard, a mouse or a voice recognition system, receives the natural-language question. Then a grammatical structure analyzer 102 analyzes the grammatical structure of the question for parsing the question into its grammatical components based on a pre-defined context-free grammatical structure. The analyzer 102 performs its tasks using a set of grammatical rules 104, and data from the database 106. Then a programming-steps generator 108 automatically generates one or more instructions based on the components. The generator 108 performs its tasks using a set of semantic rules 110 and data from the database 106. The instructions flow to a programming-steps executor 112, which executes the instructions. More than one set of instructions might be generated and executed. In at least one set of instructions, when it is executed, it queries and processes data from the database 106 for generating an answer to the question. The presenter 120, which is an output device, such as a monitor, a printer or a voice synthesizer, presents the answer to a user of the system.

FIG. 3A shows one physical embodiment 150 implementing one embodiment of the invention, preferably in software and hardware. The embodiment 150 includes a server computer 152 and a number of client computers, such as 154, which can be a personal computer. Each client computer communicates to the server computer 152 through a dedicated communication link, or a computer network 156.

FIG. 3B shows one embodiment of a client computer 154. It typically includes a bus 159 connecting a number of components, such as a processing unit 160, a main memory 162, an I/O controller 164, a peripheral controller 166, a graphics adapter 168, a circuit board 180 and a network interface adapter 170. The I/O controller 164 is connected to components, such as a hard disk drive 172 and a floppy disk drive 174. The peripheral controller 166 can be connected to one or more peripheral components, such as a keyboard 176 and a mouse 182. The graphics adapter 168 can be connected to a monitor 178. The circuit board 180 can be coupled to audio signals 181; and the network interface adapter 170 can be connected to the network 120, which can be the Internet, an intranet, the Web or other forms of networks. The processing unit 160 can be an application specific chip.

Different elements in the present invention may be in different physical components. For example, the input device 56, the presenter 120, the grammatical structure analyzer 102 and the grammatical rules may be in a client computer; and the study-materials generator 52, the question comparator 60, the database 106, the programming-steps generator 108 and the program executor 112 may reside in a server computer. In another embodiment, the database is in the server computer; and the input device 56, the study-materials generator 52, the question comparator 60, the grammatical structure analyzer 102, the programming-steps generator 108, the program executor 112 and the rules reside in a client computer. Yet in another embodiment, the embodiment 50 is in a client computer.

In this invention, the subject can be broad or narrow. In one embodiment, the subject can cover mathematics or history, or it can cover the JAVA programming language. In another embodiment, the subject covers information in a car, such as a Toyota Camry, and the user wants to understand this merchandise before buying it. In yet another embodiment, the subject covers the real estate market in a certain geographical area, and again the user wants to understand the market before buying a house.

As an example, the subject is American history. Historical facts and insights are arranged in chronological order. It starts with an introduction of the British empire before 1776, and then other information is arranged sequentially in time. In one embodiment, events happened within a certain time frame, such as one week, are grouped together as one item. And items can form a hierarchy structure. There can be a day-item, week-item, month-item and year-item. There can be long periods of time without significant events, and this leads to a month-item or a year-item.

As another example, the subject is mathematics, which is separated into major-topics, minor-topics and line-items:

Major Topics Under Mathematics
Calculus
Geometry
Trigonometry
. . .
High School Algebra
Minor Topics Under High School Algebra
Decimal Numbers
Polynomials
Linear Equations
Quadratic Equations
. . .
Integers
Line Items Under Integers
Addition & Subtraction (Int+/–)
Multiplication (Int*)
Division
Prime Numbers
Factorization
Common Divisor
. . .
Fractions
Line Items Under Fractions
Addition and Subtraction with Common Denominator (Frt+/– w/Comm Denom)
Addition and Subtraction with Integers (Frt+/–, w/Int)
Addition and Subtraction without Common Denominator (Frt+/– w/o Comm Denom)
Multiplication and Divisions with Integers (Frt*,/w/Int)
Multiplication and Division with fraction (Frt*,/)
Compound Fractions
Fraction Reduction (Frt Reduction)
Ratios and Proportions

The major-topics, minor-topics and line-items form a hierarchy tree, which shows their relationships, and their relative difficulties. Each line-item also has a number of difficulty levels ranging from 1 to 10. FIG. **4** illustrates a part of such an hierarchy for the line-items under fractions. For every path in the figure, there is a number and an alphabet in parenthesis. The number denotes the minimal difficulty level, and the alphabet denotes the minimal grade that the user has to achieve before the user is qualified to work on the subsequent line-item. For example, the user has to achieve at least a difficulty level of 8 and a grade of B for the line-item of Int+/– before the user is qualified to work on the line-item of Frt+/– w/Comm Denom. Generating study materials for each difficulty level of each line-item should be obvious to those skilled in the art.

In one embodiment, the system **50** further includes an access gate **62**. When the user wants to learn a subject, he enters his name and may be his password with the title of the subject he wants to learn through the input device **56** into the access gate **62**. The access gate **62** accesses the database to determine if he has used the system before, or if the user has used the system to learn the subject before. If he has not used the system to learn the subject before, the access gate **62** asks the study-materials generator **52** to retrieve introductory study materials on the subject for the user. In another embodiment, the subject does not have any introductory materials, and he starts the learning process by entering questions.

In yet another embodiment, the database **106** stores the questions asked by a number of prior users, and the question comparator **60** compares the questions asked by them to determine questions that are commonly-asked. Comparison processes will be described below. The term "commonly-asked" may be defined as being asked by more than 50% of the prior users, or by other metrics. The study-materials generator **52** retrieves a set of study materials answering the commonly-asked introductory question, and presents them to him. Answer-generation processes will be described below.

For different parts of the subject, again there might be one or more questions commonly asked by others. Answers to those questions can be presented to him when he starts working on those areas of the subject.

After learning the introductory material, the user may start asking questions by entering them into the system. Each question may be entered into the system **50** orally through a voice recognition input device, or through a keyboard, or other types of input device **56**. FIG. **5** shows one embodiment **190** implementing the present invention. First, the answer generator **100** retrieves (Step **192**) the question entered, and generates (Step **194**) an answer to the question. The presenter **120** then presents (Step **196**) the answer to the question, and the comparator **60** compares (Step **198**) with one or more questions previously entered by the user to determine his understanding level in the subject. The sequence of the steps of comparing (**198**) and presenting (**196**) can be changed, or the two steps can be simultaneously performed.

In one embodiment, the question just asked by the user is stored in the database **106** with his identity. In another embodiment, the database also stores a time-stamp indicating the time when the user asks the question.

There are a number of ways to generate (Step **194**) an answer to the question entered. The following description starts with answering natural-language questions that are grammatically context-free, and then extends to answering other types of questions.

A natural-language question can be in English or other languages, such as French. Examples of natural-language questions are:

Who is the first President?
What are the Bills of Right?
Where is the capital of Texas?
What is the immediate cause to the Civil War?
Why did President Nixon resign?
Who is the third President?
Who is the President after John Kennedy?
When did President Lyndon Johnson die?
When was President Nixon born?
What is the derivative of sin(x+4) with respect to?
Why is delta used in step 4 of the proof?

A statement that is not based on a natural language is a statement that is not commonly used in our everyday language. Examples are:

For Key in Key-Of(Table) do
Do while x>2

A grammatically-context-t-free question is a question whose grammar does not depend on the context. Each word in the question has its own grammatical meaning, and does not need other words to define its grammatical meaning. Hence, the grammatical structure of the question does not depend on its context. Note that "a word" can include "a number of contiguous words." This is for situations where a term includes more than one word but has only one grammatical meaning, such as the preposition "with respect to."

The question includes one or more grammatical components. A grammatical component is a component with one or

more grammatical meanings, which are defined by a set of grammatical rules to be explained below. For example, the word "president" is a noun, which has a grammatical meaning. So the word "president" is a grammatical component.

The present invention includes a database, which can be a relational database, an object database or other forms of database The database can reside in a storage medium in a client computer, or a server computer, or with part of it in the client computer and another part in the server computer.

In one embodiment, the database includes a number of tables. A table can be treated as a set of information or data grouped together that have some common characteristics. The data in each table can be further divided into different areas, and each area is represented by an attribute, which is equivalent to an identifier for a group of data that are more narrowly focused than all the data in a table. In the present invention, tables and attributes have similar function, except a table may be considered to have a broader coverage, and an attribute a narrower focus. In some examples, a table has two dimensions, as will be explained below.

Some values or data in the database may be unique. For example, if a value is a person's social security number, that value is unique. Such values are known as key values, and their corresponding attributes are known as key attributes. Note that a table can have one or more key attributes, and a key attribute may in turn be formed by more than one attribute.

One embodiment of the database **106** includes a grammatical table **114**, one or more topic-related tables **116**, and two semantic tables, **118A** and **118B**. In a general sense, the grammatical table **114** determines the grammatical meaning of each word in the question, such as whether a word is a noun or a verb. Each topic-related table **116** groups data related to a topic together in a specific format. Separated into a topic-dependent semantic table **118A** and a topic-independent semantic table **118B**, the semantic tables define the semantic meaning of each word, such as whether a word refers to an algorithm or data in a topic-related table.

The grammatical table **114** defines the grammatical meanings of words used in the natural-language question. If questions entered into the system is limited to only one subject, such as history, the grammatical table will include words in that subject, and words commonly-used by a user of the system in asking questions. Each word in the table may be defined in the following format:

```
CREATE TABLE Grammatical (
    word                Character string NOT NULL,
                        //the word
    grammatical-meaning Character string NOT NULL, //e.g.
                        "Examiner"
                        //has "noun" as its grammatical
                        meaning
)
```

Each topic-related table combines data related to a topic in a specific format. As an example, one table includes all the data related to the Presidents of the United States, and another includes all the data related to the First Ladies of the United States. The table may be two-dimensional, and include a number of columns and rows. All the data in a column or a row typically have one or more common characteristics. For example, one row includes data that identify all the bills passed by the Presidents. For a two-dimensional table, data in a row can have one characteristic, and data in a column can have another characteristic. For example, data in one column identify the heights of the Presidents, and data in a row identify data related to one specific President; the following describes an example of data alone the row:

```
CREATE TABLE PRESIDENT AS (
    //Table of all U.S. Presidents & Vice Presidents
    Name        Character string KEY, //President Name--a key attribute
        Position    Character string,  //President, Vice President
        Start_Year  Integer,           //First Year of Presidency
        End_Year    Integer,           //Last Year of Presidency
        Born_Date   Date,              //Date of Birth
        Death_Date  Date,              //Date of death
)
```

There is also a table-structure dictionary, which defines how the topic-related tables arrange their data. This dictionary is typically not considered as a part of the database. It does not contain topic-related data, but it contains structures of the topic-related tables in the database. Many database management systems automatically generate the table-structure dictionary based on the programing statements defining the topic-related tables, such as the CREATE clauses in SQL-like languages. As an example, the table-structure dictionary defines the structure of the data in the above President table by indicating that the first entry represents the name of the president, the second the position, and so on. Thus, the dictionary can contain the name of the table (the table name), the name of the table's attributes (attribute names), and their corresponding data types.

A word in the question may need one or both of the semantic tables. The topic-independent semantic table **118B** defines whether a word stands for an algorithm or data in a topic-related table. Such a table may be defined as follows:

```
CREATE TABLE Topic_Independent_Semantic (
    word        NOT NULL, //the word
    semantics,            //Indicates if the word refers to data in a
                          //topic-related table, an algorithm etc.
                          If the
                          //word is mapped to an algorithm, that
                          //algorithm will also be identified,
                          as will be
                          //further explained below.
    synonym,              //A word might have synonyms,
                          as will be
                          //further explained below.
)
```

Words with similar meaning are grouped together and are represented by one of those words as the synonym for that group of words. If a word does not have other words with similar meaning, the synonym is the word itself.

Many words do not point to an algorithm. They correspond to data in topic-related tables. The topic-dependent semantic table **118A** identifies the semantic meaning of those words through matching them to data in topic-related tables. For example, the adjective "first" applying to the President's table may operate on the data under the inauguration date attribute; on the other hand, the adjective "first" applying to the First Ladies' table may operate on the data under the date of death attribute. Such a topic-dependent table **118A** may be defined as follows:

```
CREATE TABLE Topic_Dependent_Semantic (
    Table_Name NOT NULL,  //For a table with the name
                          Table_Name:
        Who_Attribute,    //The attribute associated with 'who'
        When_Attribute,   //The attribute name associated with
                          'when'
```

-continued

```
    {i-pronoun}_Attribute,      //The attribute associated with the
                                //{i-pronoun}. The symbols { }
                                denote the
                                //word it contains. Here, the word is
                                //an i-pronoun.
    ...
    {Adj}_Attribute,
        //The attribute associated with the adjective {adj}. In this
        //example, the word is an adjective.
    {Noun}_Attribute,
        //Attribute name associated with the noun {noun}. Certain
        //nouns may refer instead to an algorithm, such as "sum."
)
```

In one embodiment, the grammatical analyzer **102**, the grammatical rules **104** and the grammatical table **114** are in a client computer. The programming-steps generator **108**, the semantic rules **110**, the semantic tables **118** and the table-structure dictionary are in a middleware apparatus, which can be a Web server. The programming-steps executor **112** with the topic-related tables are in a back-end server, which can be a database server.

One embodiment includes a computer-readable medium that encodes with a data structure including the semantic tables **118**. Another embodiment includes a computer-readable medium that encodes with a data structure including the semantic tables **118** and topic-related tables **116**. Yet another embodiment includes a computer-readable medium that encodes with a data structure including the semantic tables **118** and the grammatical table **114**. Yet a further embodiment includes a computer-readable medium that encodes with a data structure including the grammatical table **114**, the topic-related tables **116** and the semantic tables **118**.

FIG. **6** shows a set **200** of steps to implement one embodiment of the present invention. A natural-language question is entered into the answer generator **100**. The system analyzes (Step **202**) the grammatical structure of the question so as to parse it into its grammatical components, based on a pre-defined context-free grammatical structure. This task uses a set of grammatical rules **104** and the grammatical table **114**. Then, the system transforms (Step **206**) at least one component into one or more instructions using a set of semantic rules **110** with one or both of the semantic tables **118**. Then, the one or more steps are executed (Step **208**) to access and process data from one or more topic-related tables so as to generate an answer to the question.

In another embodiment, the programming-steps generator **108** transforms all the grammatical components of the question into instructions using semantic rules **110** with one or both of the semantic tables. Then the executor **112** executes all the steps to access and process data from one or more topic-related tables for generating an answer to the question.

Grammatical Structure Analyzer

In one embodiment, the analyzer **102** scans the question to extract each word in the question. Then the analyzer **102** maps each extracted word to the grammatical table **114** for identifying its grammatical meaning. For example, the word "Clinton" is identified by the grammatical table to be a proper noun; and the word "sum" is a noun. After establishing the grammatical meaning of each word, the analyzer **102** uses a set of grammatical rules to establish the grammatical components of the question based on the pre-defined context-free grammatical structure.

For a number of words, their grammatical meaning depend on their adjacent words. In one embodiment, the analyzer **102** combines each word with its contiguous words to determine its grammatical component. For example, if the word is "With," in analyzing its Grammatical meaning, the analyzer **102** identifies its contiguous words. If its contiguous words are "respect to," then the three words are combined together and are considered as one preposition. Thus, to determine grammatical meaning of a word, the analyzer identifies that word, and then a number of words following it, such as two words following it. The analyzer **102** analyzes the identified words as a unit. If the analyzer **102** cannot identify the grammatical meaning of that sequence of words, the analyzer **102** removes the last word from the sequence, and analyzes them again. The process repeats until either a grammatical meaning is found, or there is no more word. Any time when the analyzer **106** has identified a grammatical meaning, that word or sequence of words would be considered as one unit.

In one embodiment, the pre-defined context-free grammatical structure is shown in FIG. **7** and is as follows:

<Question>=<i-pronoun><aux-verb><noun-phrase> [<verb-phrase>]

where: the symbols <> denote whatever inside is a meta-symbol, which has a grammatical meaning; the meta-symbol is not in the grammatical table.

The symbols [] denote whatever inside the bracket is optional.

<I-pronoun> denotes an interrogative pronoun which is a pronoun used in asking questions, and can be one of the following: what, when, where, who, whom, whose, which, and why.

<Aux-verb> denotes an auxiliary verb, and can be any form of the verb "to be," or "do."

<Noun-phrase> is defined as <group-of-nouns> [<prepositional-noun-phrase>]

where: <group-of-nouns> is defined as:

[<modify-article>] <adjective>*<one-or-more-nouns>;

the symbol * denotes zero or more;

<modify-article> is defined as a modified article, including a, an, the, this, these and those; and

<one-or-more-nouns> denotes one or more nouns; and

<prepositional-noun-phrase> is defined as a <preposition><noun-phrase>.

<Verb-phrase> denotes a non-aux-verb, and is defined as <non-aux-verb>[<prepositional-noun-phrase>].

<Preposition> denotes a preposition defined in the grammatical table.

<Non-aux-verb> denotes a verb defined in the grammatical table and is not an <aux-verb>.

<Noun> denotes a noun defined in the grammatical table.

<Adjective> denotes an adjective defined in the grammatical table.

The pre-defined structure is only one example to illustrate the present invention. Other context-free grammatical structures are applicable also. Generating different context-free grammatical structures should be obvious to those skilled in the art.

In the present invention, a word or a set of words that can fit into the structure of a meta-symbol is a grammatical component. For example, the phrase "with respect to x" is a grammatical component, whose grammatical meaning is a prepositional-noun-phrase.

In the present invention, grammatical rules and the pre-defined grammatical structures are linked. Once the rules are set, the structures are determined. Similarly, once the struc-

tures are determined, a set of rules can be found. For example, based on the pre-defined structures, one grammatical rule is that "a group-of-nouns preceding a prepositional-noun-phrase is a noun-phrase."

The grammatical table defines the grammatical meaning of each word. In one embodiment, the table is a part of the grammatical rules. In another embodiment, all the grammatical rules that define the grammatical meaning of each word are separated from the rest of the grammatical rules, and are grouped together to establish the grammatical table **114**.

A number of examples on analyzing a question for parsing it into its grammatical components based on the pre-defined grammatical structure are:

1. What is the derivative of sin(x+4) with respect to x?

Starting from the right hand side,
- (x) is a noun
- so (x) is a group-of-nouns
- so (x) is a noun-phrase
- so (with respect to x) is a prepositional-noun-phrase
- (sin(x+4)) is a noun
- so (sin(x+4)) is a group-of-nouns
- so (sin(x+4) with respect to x) is a <group-of-nouns><prepositional-noun-phrase>
- so (sin(x+4) with respect to x) is a noun-phrase
- so (of sin(x+4) with respect to x) is a prepositional-noun-phrase
- (derivative) is a noun
- (the) is a modify-article
- so (the derivative) is a group-of-nouns so (the derivative of sin(x+4) with respect to x) is a <group-of-nouns><prepositional-noun-phrase>
- (is) is an aux-verb
- (What) is an i-pronoun

Thus, the question is of the structure
<i-pronoun><aux-verb><group-of-nouns><prepositional-noun-phrase>.

2. Why is delta used in step 4 of the proof?

Starting from the right:
- (proof) is a noun
- (the) is a modify-article
- so (the proof) is a group-of-nouns
- so (the proof) is a noun-phrase
- so (of the proof) is a prepositional-noun-phrase
- (4) is a noun
- (step) is a noun
- so (step 4) is a group-of-nouns
- so (step 4 of the proof is a <group-of-nouns><prepositional-noun-phrase>
- so (step 4 of the proof) is s noun-phrase
- so (in step 4 of the proof) is a prepositional-noun-phrase
- (used) is a verb as defined by the grammatical table
- so (used in step 4 of the proof) is a verb-phrase
- (delta) is a noun
- so (delta) is a group-of-nouns
- so (delta) is a noun-phrase
- (is) is an aux-verb
- (Why) is an i-pronoun

Thus, again the question is of the structure
<i-pronoun><aux-verb><noun-phrase><verb-phrase>.

3. Why did President Nixon resign?

Starting from the right-hand side
- (resign) is a verb
- so (resign) is a verb-phrase
- (Nixon) is a noun
- (President) is a noun
- (President Nixon) is a one-or-more-nouns
- so (President Nixon) is a noun-phrase
- (did) is an aux-verb
- (Why) is an i-pronoun

Thus, the question is of the structure
<i-pronoun><aux-verb><noun-phrase><verb-phrase>

Many questions cannot be parsed based on the pre-defined context-free grammatical structure. In this disclosure, these questions are considered as ambiguous questions, and still be analyzed through methods explained later in this disclosure. If there are more than one such pre-defined context-free granmatical structure stored in the system, the question entered will be parsed based on each structure individually. The question only has to be successfully parsed based on one such structure. If the question cannot be parsed based on all the pre-defined context-free grammatical structures, the question will be considered as an ambiguous question.

Programming-steps Generator

The programming-steps generator **108** transforms at least one grammatical component of the question using a set of semantic rules and one or both of the semantic table to generate a set of instructions. The semantic rules and the semantic tables depend on the pre-defined context-free grammatical structure, which the parsing process bases on. In one embodiment, the semantic rules are also embedded in the semantic tables. In a general sense, the generator **108** directs different grammatical components in the question to algorithms or to data in the topic-related tables.

To help explain the present invention, a number of functions are created as shown in the following:

Keys-Of(Table)
This function extracts all the key attributes in the identified table.

Attributes-Of(Table)
This function extracts all the attribute names in the identified table.

Attribute-Names({adjective}, Table)
This function identifies one or more attributes when the {adjective} is applied to the table.

Attribute-Names({noun}, Table)
This function identifies one or more attributes when the {noun} is applied to the table.

Attribute-Name({i-pronoun}, Table)
This function identifies the attribute vhen the {i-pronoun} is applied to the table.

Tables-Of({proper noun})
This function identifies one or more tables that contain the {proper noun} as a key value. It can be derived by the following program:

```
T-Names="";
for Table in {all Tables}//{all Tables} is a list of topic-related tables
do
  for Key in Keys-Of(Table)
  do
    if any value of the attribute Key in the Table contains {proper noun}
      then T-N Names=T-Names+Table
    endif
  endfor
endfor
return T-Names
```

Synonym({word})
This function identifies the synonym corresponding to the word. The synonym can be found in the topic-independent-semantic table.

Methods to create the above functions should be obvious to those skilled in the art of programming.

Based on a number of semantic rules, the programming-steps generator **108** generates instructions based on the grammatical components in the question. The following shows examples of different instructions generated to illustrate the present inventions. The instructions generated are either in a SQL-like, a LISP-like or a C-like language though other programming languages are equally applicable.

A Proper Noun

A grammatical component in the question can be a proper noun, which implies that it has a grammatical meaning of a proper noun. One set of semantic rules is that the programming-steps generator **108** transforms the proper noun into instructions to select one or more topic-related tables, and then transforms other grammatical components in the question into instructions to select and to operate on data in the tables for answering the question.

Using the topic-dependent semantic table **118A**, the programming-steps generator **108** first retrieves all tables where the proper noun is an attribute. Then, as shown in the topic-dependent semantic table, all key attributes in those tables are identified, and each of them is matched to the proper noun. The table of any key attribute that matches the proper noun is selected for additional operation by the remaining grammatical components in the question.

A proper noun may consist of more than one word, such as the "Bills of Right." A proper noun can be a lower-case word, such as "moon."

In one example, the corresponding instructions are as follows:

```
for Table in Table-Of({proper noun})
do
  for Key in Keys-Of(Table)
  do
    x=(SELECT . . .
       FROM Table
       WHERE Key MATCH {proper noun})
       // The above clause has the meaning of "where
          the key attribute
       // in the table matches the proper noun."
    if x is valid then done
    // if the SELECT function successfully identifies one
       or more attributes,
    // x is valid.
  endfor
endfor.
```

Common Nouns

One grammatical component in the question can be a common noun. The programming-steps generator **108** might transform the common noun into instructions to select a topic-related table, an attribute name, a synonym of an attribute name, the data under an attribute, or an algorithm.

As shown in FIG. **8A**, if the noun is the name of a topic-related table as shown by the topic-dependent semantic table **118A**, then the programming-steps generator transforms the noun into instructions to select (Step **252**) that topic-related table, and transforms (Step **254**) other components in the question to identify data in the table and to operate on them, if necessary.

If the noun denotes an attribute name or a synonym of an attribute name, again as shown by the topic-dependent semantic table **118A**, the programming-steps generator searches and identifies the attribute based on the noun. The instruction generated can be, for example, modifying a SELECT clause as follows

```
for Attribute in Attribute-Names({noun}, Table)
do
  SELECT Attribute from Table
  . . .
endfor
```

After all of the relevant attributes have been identified, data in them are retrieved for further processing by other parts of the question to generate an answer.

If the noun denotes the data under an attribute, the programming-steps generator identifies the data, with its corresponding attribute and table. The instructions generated can be, for example, (1) identifying each table in the function Tables-Of({noun}); (2) for each table identified, the function Attribute-Names({noun}, Table) returns the corresponding attributes containing the {noun} in that table; and (3) the remaining parts of the question operate on information under each attribute to generate the answer to the question. One set of instructions achieving such objectives is as follows.

```
for Table in Tables-Of({noun})
do
for Attribute in Attribute-Names({noun}, Table)
do
  SELECT. . .
    FROM Table
    WHERE Attribute={noun}
  . . .
endfor
  . . .
endfor
```

As shown in FIG. **8B**, the programming-steps generator might identify the algorithm (Step **262**) corresponding to the noun; the algorithm is then applied (Step **264**) to data selected by grammatical components in the question other than the common noun. For example the noun "sum" indicates accumulating results; the noun "count" indicates computing the cardinally of the results; and the noun "product" in mathematics indicates multiplying results. The topic-independent semantic table **118B** can point to locations to get the algorithm.

A Group of Nouns

If the question includes a group of nouns linked together, such as X1 X2 X3 . . . Xn, then X1 to Xn−1 can modify, the final noun Xn, which is known as the primary noun. In other words, the programming-steps generator operates on the primary noun as a common noun, or a proper noun, whichever it may be, and the remaining nouns X1 to Xn−1 further operate on data/table(s) selected by the primary noun.

Non-Auxiliary Verbs

One grammatical component can be a non-auxiliary verb. It relates to one or more events or an action, which has a number of attributes; and it might have words with similar meaning. One approach is to identify the verbs with similar meaning. Then other components in the question identify data in the attributes of the identified verbs for answering the question.

A verb can be related to many different events. As an example, the verb is "nominate": one event can be President Bush being nominated to be the President, and another event can be President Clinton being nominated to be the President.

However, an event is related to a verb. The attributes of the event can have a subject-agent, which is the agent performing the event, such as the party nominating the president. Typically, the preceding noun phrase before the verb identifies the subject-agent. The event can have an

object-agent if the verb is a transitive verb, which is the agent acted upon by the event, such as the president being nominated.

Each event has a duration that is between a starting and an ending time. For example, if the event is "walk," its duration starts with the sole of a foot changing its position from touching the ground to not touching the ground, and then ends with the sole back to touching the ground again.

Non-auxiliary verbs are grouped together in an event table, which is a topic-related table, with the topic being events. The following is an example of an event in the table:

```
CREATE TABLE EVENT (
    Verb_word        Character String NOT NULL,
                         //The verb that associates with the event
    Subject_Agent    Character String,   //Agent name performing the event
    Object_Agent     Character String,   //Agent name acted upon by the event
    Start_Time       Time,               //Starting time of event
    End_Time         Time,               //Ending time of event
    Description      Character String,   //Describes the event
    KeyId            Integer,            //Unique number identifying the event
)
```

The subject-agent, object_agent etc. are attributes related to the verb_word, which is associated with an event.

There might be non-auxiliary verbs with similar meaning as the non-auxiliary verb in the question. These verbs can be identified by the synonym in the topic-independent semantic table. As an example, the verbs of breathe and inhale have similar meaning.

As shown in FIG. **9**, the programming-steps generator **108** transforms the non-auxiliary verb in the question into one or more instructions, which select (Step **300**) one or more verbs with their attributes in the event table. The one or more verbs have similar meaning as the non-auxiliary verb. Then other components in the question identify data (Step **302**) in he attributes for answering the question. The selected verbs can be put into a temporary table or a view (a database terminology) as follows.

```
CREATE VIEW Verb_View({verb}) As
  // View is a logical table that is created only when it is needed.
  // All events matching {verb} are grouped from the event table
  // to form the view.
  SELECT*FROM EVENT
    // here * denotes all of the attributes
  WHERE Synonym({verb})=Verb_word;
```

The attributes of the selected verbs are also identified. Then, the programming-steps generator **108** generates additional instructions based on other components in the question to identify data (Step **302**) in the selected attributes for answering the question.

Events might be related. Two events may form a sequential relationship, where one event follows another event, such as eat and drink. Two events may form a consequential relationship, such as braking and stopping, with the braking event causing the stopping event. Many small events may make up a big event, with the big event containing the small events; this leads to containment relationships. Also, events may be related because they involve the same subject-agent; and events may be related because they involve the same object-agent.

An event-relationship table describes relationships among events. It can have the following format:

```
CREATE TABLE EVENT_RELATIONSHIP (
    KeyId1         Integer,          //KeyId of an event
    KeyId2         Integer,          //KeyId of another event
    Relationship   Character String,
        //Relationship, such as sequential, consequential, containment etc.
)
```

Interrogative Pronouns

Based on the interrogative pronoun in the question, the programming-steps generator **108** Generates one or more instructions to select one or more attributes in one or more tables. Those tables have been selected by grammatical components in the question other than the interrogative pronoun. The function Attribute-Name({i-pronoun}, Table) generates the attribute name corresponding to the {i-pronoun}.

One way to generate a SQL-like instruction corresponding to the {i-pronoun} is to modify a SELECT clause:

```
SELECT Attribute-Name({i-pronoun}, Table) FROM Table
```

Determiners

Examples of a set of semantic rules on determiners are:

If the determiner is "a" or "an", select any result from the previous query.

If the determiner is "some", select more than one result from the previous query. If the previous query yields only one result, that result will be selected.

If the determiner is "all", select all result from the previous query.

If the determiner is "the", modify the following SELECT function with DISTINCT, as will be shown by examples below.

Auxiliary Verbs

An auxiliary verb together with either its immediate noun phrase or a non-auxiliary verb determine whether the answer should be singular or plural.

Adjectives

One grammatical component of the question can be an adjective. As shown in FIG. **10**, based on the adjective, the programming-steps generator either identifies the value of an attribute, or identifies an algorithm. The grammatical components in the question other than the adjective have already selected one or more topic-related tables.

As shown by the topic-independent semantic table, the adjective may identify (Step **350**) an attribute. The function Attribute-Names({adjective}, table) can retrieve the attribute in the table previously selected. The corresponding instruction can be:

```
for Attribute in Attribute-Names({adjective}, Table)
do
  SELECT . . .
    FROM Table
    WHERE Attribute={adjective}
    // or "Where the attribute in the table is equal to the adjective."
  . . .
endfor
```

As an example, the noun phrase is "a red apple." The noun "apple" can be associated with a table known as FRUIT, and the Attribute-Names(red, FRUIT) yield the attribute "color." The adjective "red" is interpreted:

```
WHERE color="red."
```

If there is a sequence of such adjectives, all of them can apply to the same table. The WHERE clause would be a conjunction of the adjectives, such as:

```
WHERE
  for Attribute1 in Attribute-Names({adjective1}, Table)
  do
    for Attribute2 in Attribute-Names({adjective2},
      Table)
    do
      SELECT . . .
        FROM Table
      WHERE Attribute1=={adjective1}
      and Attribute2=={adjective2}
    endfor
  endfor
```

An adjective can refer to an algorithm, as identified by the topic-independent semantic table. Grammatical components in the question other than the component that is the adjective have selected one or more topic-related tables. As shown in the topic-independent semantic table, the adjective identifies (Step **352**) one or more attributes in those tables. Then the algorithm operates (Step **354**) on one or more data in those attributes.

As an example, the adjective is "first." The topic-independent semantic table indicates that the adjective is an algorithm sorting a list of data in ascending order; the table also identifies the data in one or more attributes in one or more topic-related tables. For each attribute identified, after sorting its data, the first value will be the result. For example, the question is "Who is the first President" The table identified is the President table. The attribute whose data are to be sorted is the "date" attribute, which identifies the time each President was elected. The instruction corresponding to the adjective "first" can be as follows:

```
for Attribute in Attribute-Names(first, Table)
do
  SELECT . . .
    FROM Table
    ORDER BY Attribute ASC
  . . .
endfor
```

The symbol ASC denotes ascending.

Similarly, if the adjective is "last," then the attribute whose data are ordered is the same, but the data are sorted in a descending manner. The corresponding instruction can be as follows:

```
for Attribute in Attribute-Names(last, Table)
do
  SELECT . . .
    FROM Table
    ORDER BY Attribute DESC
  . . .
en ifor
```

The symbol DESC denotes descending.

Another example on adjective is the word, "immediate." Its interpretation depends on the word it modifies. In one example, if the word modified is "action," the word "immediate" has the same effect as the word, "first;" if the word modified is "cause," the word "immediate" has the same effect as the word "last."

There can be a sequence of adjectives. Then, the above analysis is applied in the same order as the occurrence of the adjectives.

Preposition

One grammatical component can be a preposition. A preposition can modify its previous noun phrase or verb, such as by operating on them through an algorithm identified in the topic-independent semantic table. Under some situations, with one or more tables selected by at least one grammatical component in the question other than the component that is the preposition, the algorithm identified operates on data or values in the one or more selected tables.

Under some other situations, for example, due to the prepositions 'of' and 'in', the programming-steps generator processes the grammatical component succeeding the preposition before the grammatical component preceding.

For another example, the preposition 'before' can modify the WHERE clause with a comparison on time:

{time of preceding event}<{time of succeeding event}

Programming-Steps Executor

The executor **112** executes at least one set of instructions generated from one grammatical component to at least access data from the database to generate an answer for the question, if there is one.

In one embodiment, after the programming-steps generator **108** generates a set of instructions, the programming-steps executor **112** executes them. The set may be generated from one grammatical component. This process repeats until all sets are generated and executed to answer the question. For at least one set of instructions, the executor **112** accesses data from one or more topic-related tables identified by the instructions. In another embodiment, all the instructions are generated; then the program executor **112** runs the instructions, which include accessing data from one or more topic-related tables identified by the instructions, and processing those data for generating the answer to the natural-language question.

In the appendix, there are a number of examples of instructions illustrating the present invention. They generated to answer different types of grammatically-context-free questions.

Ambiguous Questions

In the present invention, the grammatical structure analyzer **102** may decide that the natural-language question cannot be parsed into grammatical components based on the pre-defined context-free grammatical structure. For example, the grammatical components of the question cannot fit into the pre-defined structure. Then the question is considered ambiguous, and an answer cannot be generated.

Ambiguity may be due to a number of reasons. For example, the question may contain words with non-unique grammatical meaning the question may contain words not in the grammatical table, or the grammatical structure of the question is different from the pre-defined grammatical structure. FIG. **11** shows different approaches to resolve the ambiguity.

The grammatical structure analyzer can decide that a word can be of more than one grammatical meaning, such as it can be a noun and a verb. In one embodiment, the analyzer produces (Step **402**) an answer for each meaning and ignores those meaning with no answer. In another embodiment, the analyzer asks (Step **400**) the user to identify the correct grammatical meaning.

For example, the question is: "When was the Persian Gulf war?" The word "war" can be a noun or a verb. In one embodiment, the analyzer asks the user whether the word "war" is a noun or a verb. Based on the user's response, the question is analyzed. In another embodiment, the analyzer generates answers to both the question that treats the word "war" as a verb, and the question that treats the word "war" as a noun. Both answers, if available, are presented to the user.

If the grammatical structure analyzer decides that the question contains one or more words not in the grammatical table, in one embodiment, the analyzer removes (Step **404**) the un-recognized word and processes the remaining words

in the question. In another embodiment, the analyzer asks (Step **406**) the user for a different word. The analyzer might assume that the word is miss-spelled, and ask the user to correct it; the analyzer might replace (Step **408**) the un-recognized word with a word in the grammatical table most similar to or with minimum number of different characters from the un-recognized word. The analyzer then presents (step **410**) the matched word to the user to ask if that is the right word. A list of matched words may be presented for the user to select.

For example, the question is: "What exactly are the Bills of Right?"The word "exactly" is an adverb and is not in the grammatical table. The word is dropped, and the question, satisfying the grammatical structure, is analyzed. In another example, the question is: "What is the Bill of Right?" Here, the "Bill of Right" should be the "Bills of Right." The analyzer can ask the user to spell the "Bill of Right" again; or the analyzer can find the term closest in spelling to the un-recognized term, and identify the term to be the "Bills of Right". The identified word is presented to the user to ask if that is the right spelling.

In the present invention, the grammatical structure of the question entered may be different from the one or more pre-defined context-free grammatical structures in the system.

In one embodiment, a non-essential grammatical component is missing from the question. A grammatical component is non-essential if that grammatical component can be removed from the question without changing the answer to the question. For example, an auxiliary verb in certain condition can be non-essential. One approach to solve this problem is to ignore (Step **412**) the missing grammatical component in generating the answer to the question. Another approach is to add the missing non-essential grammatical component back into the question, and present to the user asking if that is correct. For example, the question is: "When President Nixon resign?" An auxiliary verb is expected after the word "When"; such a word is entered into the question, which is then submitted to the user for approval.

In another embodiment, the user is suggested to re-enter (Step **414**) the question with advice as to the appropriate question structure. One advice is to ask the user to re-enter the question based on the pre-defined structure, such as using one of the i-pronouns in the pre-defined grammatical structure. This can be done, for example, by citing a list of acceptable i-pronouns, and a list of model questions using the i-pronouns as examples. Another advice is to identify nouns and non-auxiliary verbs, if any, in the question, and to ask the user which of the identified word or words he wants to know more about. Then it would be up to the user to select the one he wants. In a farther embodiment, the identified word or words are fit into alternative grammatical structures, and the user is asked to select one structure out of the list of suggested structures.

As an example, the question is: "Do you know when President Nixon resign?" Assume that such a question does not fit the pre-defined grammatical structure. The user is suggested to re-enter the question using one of the following i-pronouns: What, when, where, why and who. In another embodiment, the noun and the auxiliary verb are identified, and they are "President Nixon resign." The user is asked, "You want to know about President Nixon resign?" In yet another embodiment, the identified words are fit into the following question formats, and it would be up the user to select one, for example:

What does President Nixon resign?

When does President Nixon resign?

Where does President Nixon resign?

Why does President Nixon resign?

Who does President Nixon resign?

As another example, the question is: "Is there a reason why President Clinton sent troops to Bosnia?" Assume that the question does not fit the pre-defined grammatical structure. In one embodiment, the user is suggested to re-enter the question using one of the i-pronouns in the pre-defined grammatical structure. In another embodiment, the nouns and the non-auxiliary verbs, "President Clinton", "troops" "send" and "Bosnia" are identified. Then the user is asked to select one or more of the following questions:

Do you want to know about President Clinton?

Do you want to know about troops?

Do you want to know about Bosnia?

Also, the answer generator **100** can present suggestions to the user on ways to rephrase the original question based on the noun and the non-auxiliary verbs. It would then be up to the user to select the one he wants.

In certain situation, the present invention does not have any answer. As an example, the grammatical table does not have some essential terms X in the question. Then, the present invention can return the following message:

> Sorry, I do not know anything about X. You may want to check with your instructor. If you wish, I will inform your instructor your question. {Click here to inform your teacher}

If the user clicks at the designated area, his last question will be automatically sent as an electronic mail to an instructor who can answer the user directly. This instructor may be previously selected by the user, or the guardian of the user.

Questions Matching Engine

The embodiment shown in FIG. **2** can answer an infinite number of questions.

FIG. **12** illustrates another embodiment **500** of the answer generator, which provides answers to a finite number of questions, but requires fewer steps to generate answers as compared to the embodiment shown in FIG. **2**. Also, the answer generator **500** can answer non-natural-language questions, and grammatically-context-t-dependent questions. In this embodiment, the database **106** includes a questions table, which contains many questions, each with its corresponding answer. A question matching engine **529** compares the question entered with questions in the database. An answer retriever **539** retrieves the answer to the question in the database that matches the entered question.

FIG. **13** shows one set of steps **550** for the present embodiment. The question matching engine **529** compares (Step **552**) the entered question with questions in the database **106**. If there is a match with any one of them, the answer retriever **639** retrieves (Step **54**) the answer corresponding to the matched question. If no question in the table matches the input question, the answer generator **500** might use one of the approaches discussed in the ambiguous questions section to answer the question.

In another embodiment, the question entered is a natural-language question. The matching engine **529** compares the grammatical components of the natural language question with components of the questions in the database **106**.

A further embodiment includes an essential-components extractor, which extracts essential components from the natural-language question entered. Only essential components are compared with the pre-stored questions, which have essential components. If there is a match, the answer to the corresponding matched question is retrieved and is presented to the student. As an example, an auxiliary verb is

a non-essential components. The extractor strips off the auxiliary verb from the question to allow the matching engine **529** to compare the rest of the components.

In yet another embodiment, the question entered is a grammatically context-free question.

The answer generator **100** shown in FIG. **2** can be combined with the answer generator **500** shown in FIG. **12**. The question entered is first analyzed by the answer generator **100** shown in FIG. **2**. If the question cannot be parsed into its grammatical components based on the pre-defined structure, then the question is passed to the answer generator **500** shown in FIG. **12**. If that answer generator also cannot find a match in the questions table, the question will be considered as an ambiguous question to be resolved by approaches discussed in the ambiguous questions section.

Question Comparator

In one embodiment, the comparator **60** compares the question just entered with one or more questions previously entered by the user to determine his understanding level in the subject. This can be done for example by the comparator **60** comparing the grammatical components of the questions. In one embodiment, non-essential components are de-emphasized. Two questions are considered identical if their essential components are identical. Words are considered identical to its synonyms, as defined by the topic-independent-semantic table in the database. If the user has asked the same question more than once, his understanding level is low in the areas covered by the question. The more times he asked the same question, the less he understands the area covered by the question.

In another embodiment, the comparator **60** counts the total number of occurrence of every interrogative pronoun, every noun and every non-auxiliary verb in the question just asked based on all the questions he previously asked. If the questions are:

Just entered: What is the derivative of sin(x+4) with respect to x?

Previously asked: What is the derivative of cos(2*x)*sin (x+4) with respect to x?

the comparator **60** has the following word counts:

what: twice,

derivative: twice,

sin: twice,

x: 4 times.

The noun x is known as an indeterminant, which is a non-essential word; it is not essential to determine his understanding level. In one embodiment, they are ignored in word counts.

If the questions are:

Just asked: When did President Clinton become president?

Previously asked: How many terms have President Clinton served?

the comparator **60** has the following word counts:

When: once,

President Clinton: twice,

president: once,

become: once.

There is also a question count for the question just asked. That count sums the word counts of the words in the question, and divides that sum by the number of essential words in the question:

Question count=Sum (Word counts)/(# of essential words in the question)

The division normalizes the question count.

Based on the above metrics, the user's understanding level in the area covered by the question is low if the question has a high question count.

In another embodiment, the word count and the question count also consider time as a factor. The user might have asked a question similar to one he just asked long time ago. In order for the word count and the question count to reflect his degree of forgetfulness, the system uses an effective word count, an effective question count, and time-stamps. The effective word count adjusts the word count by a time factor. One equation for the effective word count of a word is:

Effective word count=1+(word count)*c/exp(Current-time-stamp−Last-time-stamp),

where:

c is a positive constant between 0 and 1;

exp is the exponential function,

time-stamps are measured in units of time,

such as every ten minutes is one unit,

but with the units removed in the equation;

current-time-stamp denotes the time

when the user just asked the question with the word; and

last-time-stamp denotes the time

when the user asked a question with the word immediately before the current-time-stamp.

The effective question count sums the effective word counts of the essential words in the question, and divides that sum by the number of essential words in the question.

Effective question count=Sum (Effective word counts)/(# of essential words in the question)

The division normalizes the effective question count.

Again, based on the above metrics, the user's understanding level in the area covered by the question is low if the question has a high effective question count.

In a further embodiment, the comparator **60** also includes a word-significance table, which indicates the significance of words used in a question. Every word in the subject has a significance factor ranging from 0 to 1. For example, the non-essential components, just like the indeterminants in mathematics, have a significance factor of 0; and the interrogative pronoun "why" has a higher significance factor relative to the interrogative pronoun "what." In one embodiment, before the comparator **60** sums the word counts to generate the question counts, each word count is multiplied by its corresponding significance factor.

In another embodiment, based on the magnitude of the question count, the comparator **60** may test the user. The test results further indicate the user's understanding level in areas covered by the question. Generating a test in a certain area should be obvious to those skilled in the art and will not be further described.

In yet another embodiment, based on the user's understanding level, the comparator **60** sends a message to the study-materials generator **52** to retrieve study materials for him. In one approach, the less he understands a certain area, the more detailed is the study materials to be presented to him. In another approach, the less he understands a certain area, the lower the level of difficulty is the study materials to be presented to him. For example, if the user is very weak in fractions, then the presenter **120** presents study materials on level 1 of fractions to him. Generating and retrieving study materials with different degrees of difficulties and different amount of detail should be obvious to those skilled in the art, and will not be further described.

If the user still asks the same question after the system has presented to him detailed study materials, the answer gen-

erator **100** may ask him to consult an instructor. In one embodiment, the database **106** contains a list of instructors for different areas of the subject. With permission from the user, the answer generator **100** may contact one or more instructors through electronic mail or other means, with the question sent to the instructor. The instructor can contact the user directly.

After reading the answer to his question, the user might ask another question, and the process of answering question repeats.

Filling Gaps of Misunderstanding

This invention is also applicable to filling gaps of misunderstanding when the user is working on a subject. FIG. **14** shows one such embodiment **600**. First, the system **50** generates (Step **602**) study materials on a subject, and presents (Step **604**) the study materials to him. The subject can be a novel he wants to read on the system. The novel may have many individuals. He gets confused on their names, and begins to lose interest in the novel. Clarifying the identities might revive his interest in the novel. Similarly, the subject can be finance, and he forgets the meaning of a term, such as capital asset pricing model. He can ask the system a question. The system **50** retrieves (Step **606**) the question entered, and generates (Step **608**) an answer to the question. The presenter **120** presents (Step **610**) the answer to him. Filling gaps of misunderstanding is very important in learning. Typically, the user gets confused and loses interest in the study materials as misunderstanding increases. After the system has answered his question, the process repeats with the system **50** generating study materials for the user. In one embodiment, the process continues until the user has mastered the entire subject.

In one embodiment, after working on the subject for some time, the user stops. The database stores the time he stops, with his identity, and the location where he terminates learning the subject. Next time, when the users enters the answer generator **100** to learn the same subject again, the answer generator **100** re-starts the process from where he ended last time. In another embodiment, the answer generator **100** asks him if he wants to re-start from where he ended or to re-start from another part of the subject. It would be up to him to decide.

Other embodiments of the invention will be apparent to those skilled in the art from a consideration of this specification or practice of the invention disclosed herein. It is intended that the specification and examples be considered as exemplary only, with the true scope and spirit of the invention being indicated by the following claims.

What is claimed is:

1. A computer-implemented method for providing a user with information, said method comprising:

sending informational materials from a server system to a client system via the Internet so as to inform a user of the client system about a subject;

subsequently receiving, at the server system, a request to respond to a natural-language question asked by the user, the natural language question being asked by the user at the client system, and the natural language question being associated with the subject;

determining, at the server system, a response to the natural-language question, said determining operating to at least analyze the natural-language question using at least grammatical and semantic processing; and

sending the response to the natural-language question from the server system to the client system via the Internet;

wherein the subject relates to a product that the user is interested in.

2. A computer-implemented method as recited in claim **1**, wherein said method further comprises:

identifying additional informational materials related to the natural-language question; and

sending the additional informational materials from the server system to the client system via the Internet.

3. A computer-implemented method as recited in claim **2**, wherein said sending of the additional informational materials is performed when requested by the user.

4. A computer-implemented method as recited in claim **2**, wherein said sending of the additional informational materials is automatically performed substantially simultaneous with said sending of the response.

5. A computer-implemented method as recited in claim **1**, wherein the grammatical and semantic processing uses at least one grammatical rule and at least one semantic rule.

6. A computer-implemented method as recited in claim **1**, wherein said method further comprises:

receiving a referral request from the user requesting that the natural language question be referred to a human representative; and

enabling a human representative to respond to the natural language question.

7. A computer-implemented method as recited in claim **6**, wherein the natural language question is sent to the human representative without the need for the user to re-enter the question into the client system.

8. A computer-implemented method as recited in claim **6**, wherein the human representative is specifically for the subject, and a different human representative may be enabled for a different subject.

9. A computer-implemented method as recited in claim **1**, wherein said method further comprises:

referring the natural language question to a human representative to respond to the natural language question.

10. A computer-implemented method as recited in claim **9**, wherein the natural language question is sent to the human representative without the need for the user to re-enter the question into the client system.

11. A computer-implemented method as recited in claim **9**, wherein the human representative is specifically for the subject, and a different human representative may be enabled for a different subject.

12. A computer-implemented method as recited in claim **1**, wherein said method further comprises:

determining whether clarification of the natural language question is desirable; and

prompting the user to clarify the natural language question when clarification of the natural language question is determined to be desirable.

13. A computer-implemented method as recited in claim **12**, wherein the user is asked one or more questions when prompting the user to clarify.

14. A computer-implemented method as recited in claim **1**, wherein said method further comprises:

sending at least two questions to the client system; and

receiving, at the server system, at least an indication of a selection by the user of one of the questions sent by the server system to be responded.

15. A computer-implemented method as recited in claim **1**, wherein the response is an answer to the natural-language question.

16. A computer-implemented method as recited in claim **1**, wherein said determining operates to transform at least a portion of the natural-language question into at least one instruction.

17. A computer-implemented method as recited in claim 16, wherein the server system has an information database coupled thereto, and wherein the instruction is a query for the database.

18. A computer-implemented method as recited in claim 1, wherein said determining is independent of at least one word in the natural-language question, and the response depends on a correct spelling of at least one misspelled word in the natural-language question.

19. A computer-implemented method as recited in claim 1, wherein the subject relates to a product the user is interested in obtaining.

20. A computer-implemented method as recited in claim 1, wherein the subject relates to a product the user is interested in using.

21. A computer-implemented method as recited in claim 1, wherein said method further comprises having at least a second response available for the user, and allowing the user to select one or more of the responses for the natural-language question.

22. A computer-implemented method as recited in claim 1, wherein said determining produces a plurality of responses, and

wherein said sending operates to send the plurality of responses from the server system to the client system.

23. A computer-implemented method as recited in claim 22, wherein at least one of the responses is a question.

24. A computer-implemented method as recited in claim 22, wherein said method further comprises:

selecting one of the plurality of responses for which a further response is desired.

25. A computer-implemented method as recited in claim 24, wherein at least one of the responses is a question.

26. A computer-implemented method as recited in claim 1, wherein said method further comprises considering at least one additional input entered by the user in determining the response.

27. A computer-implemented method as recited in claim 26, wherein the additional input is a response to an inquiry from the server system.

28. A computer-implemented method as recited in claim 27, wherein the additional input is for clarifying the natural language question.

29. A computer-implemented method as recited in claim 26, wherein the additional input is also a question from the user.

30. A computer-implemented method as recited in claim 1, wherein the server system keeps track of a unique identifier of the user.

31. A computer-implemented method as recited in claim 1, wherein said method further comprises:

adapting in real-time to respond to questions by considering at least one additional input.

32. A computer-implemented method for providing a user with information, said method comprising:

sending informational materials from a server system to a client system via the Internet so as to inform a user of the client system about a subject;

subsequently receiving, at the server system, a request to respond to a natural-language question asked by the user, the natural language question being asked by the user at the client system, and the natural language question being associated with the subject;

determining, at the server system, a response to the natural-language question, said determining operating to at least analyze the natural-language question using at least grammatical and semantic processing;

sending the response to the natural-language question from the server system to the client system via the Internet;

receiving a referral request from the user requesting that the natural language question be referred to a human representative; and

enabling a human representative to respond to the natural language question,

wherein the subject relates to a product that the user is interested in using.

33. A computer-implemented method as recited in claim 32, wherein said method further comprises having at least a second response available for the user, and allowing the user to select one or more of the responses for the natural-language question.

34. A computer-implemented method as recited in claim 32, wherein said determining is independent of at least one word in the natural-language question, and the response depends on a correct spelling of at least one misspelled word in the natural-language question.

35. A computer-implemented method as recited in claim 32, wherein said determining operates to transform at least a portion of the natural-language question into at least one instruction.

36. A computer-implemented method as recited in claim 35, wherein the server system has an information database coupled thereto, and wherein the instruction is a query for the database.

37. A computer-implemented method as recited in claim 32, wherein said method further comprises:

sending at least two questions to the client system, and

receiving, at the server system, at least an indication of a selection by the user of one of the questions sent by the server system to be utilized as the natural language question being asked by the user.

38. A computer-implemented method as recited in claim 32, wherein said method further comprises considering at least one additional input entered by the user in determining the response.

39. A computer-implemented method as recited in claim 38, wherein the additional input is a response to an inquiry from the server system.

40. A computer-implemented method as recited in claim 38, wherein the additional input is also a question from the user.

41. A computer-implemented method as recited in claim 32, wherein the server system keeps track of a unique identifier of the user.

42. A computer-implemented method for providing a user with information, said method comprising:

sending informational materials from a server system to a client system via the Internet so as to inform a user of the client system about a subject;

subsequently receiving, at the server system, a request to respond to a natural-language question asked by the user, the natural language question being asked by the user at the client system, and the natural language question being associated with the subject;

determining, at the server system, a response to the natural-language question, said determining operating to at least analyze the natural-language question using at least grammatical and semantic processing;

sending the response to the natural-language question from the server system to the client system via the Internet;

identifying additional informational materials related to the natural-language question; and